STATISTICAL LEARNING FOR ANOMALY DETECTION IN CLOUD SERVER SYSTEMS A MULTI-ORDER MARKOV CHAIN FRAMEWORK

GRANDHI VIJAYA DURGA RAO
PG Scholar, Department of M.C.A,
S.K.B.R P.G College,
Amalapuram, E.G.Dt., A.P, India.
vijaygrandhi860@gmail.com

Mr. NAGA. SRINIVASA RAO*
Asst. Professor, Dept of M.C.A,
S.K.B.R P.G College,
Amalapuram, E.G.Dt., A.P, India.
naagaasrinu@gmail.com

Abstract:

As a major strategy to ensure the safety of IT infrastructure, anomaly detection plays a more important role in cloud computing platform which hosts the entire applications and data. On top of the classic Markov chain model, we proposed in this paper a feasible multi-order Markov chain based framework for anomaly detection. In this approach, both the high-order Markov chain and multivariate time series are adopted to compose a scheme described in algorithms along with the training procedure in the form of statistical learning framework. To curb time and space complexity, the algorithms are designed and implemented with non-zero value table and logarithm values in initial and transition matrices. For validation, the series of system calls and the corresponding return values are extracted from classic Defense Advanced Research Projects Agency (DARPA) intrusion detection evaluation data set to form a two-dimensional test input set. The testing results show that the multi-order approach is able to produce more effective indicators: in addition to the absolute values given by an individual single-order model, the changes in ranking positions of outputs from different-order ones also correlate closely with abnormal behaviours

keywords : Markov processes, Training, Cloud computing, Algorithm design and analysis, Mathematical model, Equations, Servers

1. INTRODUCTION

1.1 Introduction:

An increasing number of academicals and industrial users are starting to rely solely on cloud computing servers that host entire applications and storage. In fact, these cloud computing and services in the form of distributed and open structure become obvious targets for potential threats. Thus, taking care of both business and personal data, servers expose critical safety and availability issues. Their invulnerability is of major importance to both individuals and the society. However, during catastrophic disasters such as intrusion, crash or breakdown, the anomalies must be first discovered before any actual remedy could come to its aid. Being recessive at the early stage, such problems would not exhibit distinct traits and often lead to delayed responses and irrecoverable results.

Fortunately, a server is the ideal instance whose behavior manifests regularity statistically. This lays the foundation for any anomaly detection algorithm based on machine-learning or data-mining. All of them adopt the idea of extracting the patterns within the (massive) training set and thus raise the alarm on the deviate ones. In the literature, anomaly or intrusion detection was implemented in great variety of approaches. These approaches are usually categorized into three groups, i.e. statistical approaches, machine learning approaches and data mining approaches: In statistical approaches, anomaly or intrusion detection systems usually watch behaviors of observed objects to comprise statistical distributions as a set of trained profiles during the training phase. These systems then apply the set of trained profiles by comparing them against a new set of profiles of observed objects during the detection phase. An anomaly or intrusion is detected if these two sets of profiles do not match. In general, any incident whose occurrence frequency goes beyond standard deviations from statistical normal ranges raises an alarm.

Machine learning based approaches tend to reduce the supervision costs during the training phase of statistical approaches by enabling machine learning based systems to learn and improve their performance on their own. These systems are usually designed as a framework that can improve its performance in a loop cycle by adapting its execution strategies according to execution feedbacks, e.g. system call sequence analysis, Bayesian network and Markov model execution results. Neural networks and Hidden Markov Model have been proved to be useful techniques.

Data mining based approaches exploit unknown rules and patterns by exploring large amounts of data collected either online or off line. Anomaly or intrusion detection systems can be improved with additional inputs in terms of hidden patterns, associations, changes, and events found in data. Common technologies involved in data mining approaches include classification, clustering and outlier detection, and association rule discovery. Application of typical data-mining algorithms such as K-nearest neighbour or clustering could be found.

In the context of this work, we focus on approaches in categories of statistical and machine learning based approaches. Theories of the classic Markov model are applied so as to detect anomalous patterns in the system, using the ordering property of events as proposed. Ju and Vardi introduce the high-order Markov chain as an extension. Several approaches are then introduced to overcome the formidable cost that seems highly likely to come with it, including the hybrid model or support vector machine. In additional to statistical patterns, as originally proposed, the time series of system calls are now by common consent a powerful tool in identifying the nature of a system's behavior. Due to system calls' privileges, a large number of researches of intrusion detection, including and are based on exploiting, modelling, or learning from the audit data of system calls.

In 1998, the Cyber Systems and Technology Group of MIT Lincoln Laboratory conducted a seven-week simulation of intrusion in the background of daily usage, and then released all their data named as classic DARPA Intrusion Detection Evaluation Data Set. The major contribution of the paper is our approach based on multi-order Markov chains, which reveals that the combination of mixed-order Markov chains would bring considerably interesting and substantial improvement over any single-order one with fairly reasonable cost.

Utilizing not only the multiple order property, this approach effectively suits the application of anomaly detection in addition to its first practice in rainfall modeling. In our practice, the relative ranking positions between probabilities from multi-order models serve as a new effective indicator for anomalies, which refers to our finding that the ascending order suggests normal, while the descending one exhibits anomalous. Our approach differs from a recent model, which exploits mixture of Markov chains by incorporating n-gram transitions to model the normal behavior of users' HTTP requests rather than system calls in underlying servers. Secondly, we take into accounts a new category of inputs (the return values of system calls) to improve the effectiveness of the multi-order Markov chain based approach and form a two-dimensional model. In the application of anomaly detection, the conventional notion of using system calls to identify a system's behavior is insufficient in that it does not take into account the resulting status of execution. In a few more recent works, return values of system calls were to taken into consideration to detect or interpret the anomalies

1.2 Purpose:

Cloud computing comes with indispensable dependency on networked computer systems. Unfortunately, while every one knows there is no guarantee of its well being, we tend to simply ignore this painful idea. An increasing number of academicals and industrial users are starting to rely solely on cloud computing servers that host entire applications and storage. In fact, these cloud computing and services in the form of distributed and open structure become obvious targets for potential threats. Thus, taking care of both business and personal data, servers expose critical safety and availability issues. Their invulnerability are of major importance to both individuals and the society

1.3 Scope:

The series of system calls and the corresponding return values are extracted from classic Defense Advanced Research Projects Agency (DARPA) intrusion detection evaluation data set to form a two-dimensional test input set. The testing results show that the multi-order approach is able to produce more effective indicators: in addition to the absolute values given by an individual single-order model, the changes in ranking positions of outputs from different-order ones also correlate closely with abnormal behaviours.

1.4 Motivation:

We focus on approaches in categories of statistical and machine learning based approaches. In theories of the classic Markov model are applied so as to detect anomalous patterns in the system, using the ordering property of events as proposed . Ju and Vardi introduces the high-order Markov chain as an extension. Several approaches are then introduced to overcome the formidable cost that seems highly likely to come with it, including the hybrid model i or support vector machine.

1.5 Overview:

In statistical approaches, anomaly or intrusion detection systems usually watch behaviors of observed objects to comprise statistical distributions as a set of trained profiles during the training phase. These systems then apply the set of trained profiles by comparing them against a new set of profiles of observed objects during the detection phase. An anomaly or intrusion is detected if these two sets of profiles do not match. In general, any incident whose occurrence frequency goes beyond standard deviations from statistical normal ranges raises an alarm.

2. LITERATURE SURVEY

A Program based anomaly intrusion detection scheme using multiple detection engines and fuzzy inference

This work proposed a hybrid anomaly intrusion detection scheme using program system calls is proposed. In this scheme, a hidden Markov model (HMM) detection engine and a normal database detection engine have been combined to utilize their respective advantages. A fuzzy-based inference mechanism is used to infer a soft boundary between anomalous and normal behavior, which is otherwise very difficult to determine when they overlap or are very close. To address the challenging issue of high cost in HMM training, an incremental HMM training with optimal initialization of HMM parameters is suggested.

Alert correlation in collaborative intelligent intrusion detection systems—A survey

As complete prevention of computer attacks is not possible, intrusion detection systems (IDSs) play a very important role in minimizing the damage caused by different computer attacks. There are two intrusion detection methods: namely misuse- and anomaly-based. A collaborative, intelligent intrusion detection system (CIIDS) is proposed to include both methods, since it is concluded from recent research that the performance of an individual detection engine is rarely satisfactory. In particular, two main challenges in current collaborative intrusion detection systems (CIDSs) research are highlighted and reviewed: CIDSs system architectures and alert correlation algorithms. Different CIDSs system, architectures are explained and compared. The use of CIDSs together with other multiple security systems raises certain issues and challenges in, alert correlation. Several different techniques for alert correlation are discussed. The focus will be on correlation of CIIDS alerts. Computational, Intelligence approaches, together with their applications on IDSs, are reviewed. Methods in soft computing collectively provide understandable and autonomous solutions to IDS problems.

Interpreting chance for computer security by viterbi algorithm with edit distance

This work addresses the importance of chance discovery in computer security. There are various methods to discover chances in computer usage, but they have such drawbacks as discovering only anomalies, not interpreting anomalies in conventional approach in the community of computer security. This work focuses on a role of interpreting the type of anomalies by analyzing the state sequences using Viterbi algorithm and evaluating the distance between the standard model of anomaly type and the state sequence of discovered anomalies. Because the state sequences are not always extracted consistently due to environmental factor, the edit distance is utilized to measure the distance effectively.

A multi-order markov chain based scheme for anomaly detection

This work presents a feasible multi-order Markov chain based scheme for anomaly detection in server systems. In our approach, both the high-order Markov chain and multivariate time series are taken into account, along with the detailed design of training and testing algorithms. To evaluate its effectiveness, the Defense Advanced Research Projects Agency (DARPA) Intrusion Detection Evaluation Data Set is used as stimuli to our model, by which system calls and the corresponding return values form a two-dimensional input set. The calculation result shows that this approach is able to produce several effective indicators of anomalies. In addition to the absolute values given by an individual single-order model, we also notice a novelty unprecedented before, i.e., the changes in ranking positions of outputs from different-order ones also correlate closely with abnormal behaviours.

Spectrogram: A mixture-of-Markov-chains model for anomaly detection in web traffic

This work proposed a Spectrogram, machine learning based statistical anomaly detection (AD) sensor for defense against web-layer code-injection attacks. These attacks include PHP file inclusion, SQL-injection and cross-site- scripting; memory-layer exploits such as buffer overflows are addressed as well. Statistical AD sensors offer the advantage of being driven by the data that is being protected and not by malcode samples captured in the wild. While models using higher order statistics can often improve accuracy, trade-offs with false-positive rates and model efficiency remain a limiting usability factor. This work presents a new model and sensor framework that offers a favorable balance under this constraint and demonstrates improvement over some existing approaches. Spectrogram is a network situated sensor that dynamically assembles packets to reconstruct content flows and learns to recognize legitimate web-layer script input. This work describes an efficient model for this task in the form of a mixture of Markov- chains and derives the corresponding training algorithm.

Modelling rain risk: A multi-order Markov chain model approach

The purpose of this work is to investigate the best frequency description of a chain dependent Markov process for the daily simulation of precipitation. The influence of the order of the Markov chain model to simulate daily precipitation occurrence is evaluated. A mixed-order model is constructed and compared to a simple first-order model to evaluate the importance of the model order for the pricing of a rainfall index put option. Design/methodology/approach–For the first time a mixed-order Markov chain model is presented where the monthly varying order was chosen based on a Bayesian information criteria analysis of rainfall data for one weather station in the US. The outcome of this model is compared to simpler Markov models and to burn analysis results. Findings–The comparison indicate that there is only a slightly better representation of the rain statistics in the theoretically best mixed-order Markov chain model compared to a simpler first-order model. Clear differences between the daily simulation and the burn method are found when pricing a put option on a rainfall index. All daily simulation models underestimate the volatility of the monthly rainfall amount especially in the summer months. Research limitations/implications–To assesses the robustness and any geographical dependence of the bias in the volatility a systematic analysis could be applied to more weather stations across the US in further studies. Practical implications–The bias in the volatility has significant influence on the price of the put option considered here and limits the use of such a model for risk analyses, e.g. for an extreme event cover. Originality/value–For the first time a multi-order Markov chain model is applied to price a precipitation derivative.

Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation

An intrusion detection evaluation test bed was developed which generated normal traffic similar to that on a government site containing 100's of users on 1000's of hosts. More than 300 instances of 38 different automated attacks were launched against victim UNIX hosts in seven weeks of training data and two weeks of test data. Six research groups participated in a blind evaluation and results were analyzed for probe, denial of -service (DoS), remote-to-local (R2L), and user to root (U2R) attacks. The best systems detected old attacks included in the training data, at moderate detection rates ranging from 63% to 93% at a false alarm rate of 10 false alarms per day. Detection rates were much worse for new and novel R2L and DoS attacks included only in the test data. The best systems failed to detect roughly half these new attacks which included damaging access to root-level privileges by remote users.

Profiling program behavior for anomaly intrusion detection based on the transition and frequency property of computer audit data

Intrusion detection is an important technique in the defense-in-depth network security framework. In recent years, it has been a widely studied topic in computer network security. In this paper, we present two methods, namely, the Hidden Markov Models (HMM) method and the Self Organizing Maps (SOM) method, to profile normal program behavior for anomaly intrusion detection based on computer audit data. The HMM method utilizes the transition property of events while SOM method relies on the frequency property of events. Two data sets, CERT synthetic Sendmail system call data collected in the University of New Mexico (UNM) and Live FTP system call data collected in the CNSIS lab of Xi'an Jiaotong University, were used to assess the two methods. Testing results show that the HMM method using the transition property of events produces good detection performance while high computational expense is required both for training and detection. The HMM method is better than other two methods reported previously in terms of detection accuracy for the same data set. The SOM method considering the frequency property of events, on the other hand, is suitable for real-time intrusion detection because of its capability of processing a large amount of data with low computational overhead

3. EXISTING SYSTEM

In existing, anomaly or intrusion detection was implemented in great variety of approaches. These approaches are usually categorized into three groups, i.e. statistical approaches, machine learning approaches and data mining approaches: In statistical approaches, anomaly or intrusion detection systems usually watch behaviors of observed objects to comprise statistical distributions as a set of trained profiles during the training phase. These systems then apply the set of trained profiles by comparing them against a new set of profiles of observed objects during the detection phase. An anomaly or intrusion is detected if these two sets of profiles do not match. In general, any incident whose occurrence frequency goes beyond standard deviations from statistical normal ranges raises an alarm. Machine learning based approaches tend to reduce the supervision costs during the training phase of statistical approaches by enabling machine learning based systems to learn and improve their performance on their own. These systems are usually designed as a framework that can improve its performance in a loop cycle by adapting its execution strategies according to execution feedbacks, e.g. system call sequence analysis, Bayesian network and Markov model execution results. Neural networks and Hidden Markov Model have been proved to be useful techniques. Data mining based approaches exploit unknown rules and patterns by exploring large amounts of data collected either online or off line. Anomaly or intrusion detection systems can be improved with additional inputs in terms of hidden patterns, associations, changes, and events found in data. Common technologies involved in data mining approaches include classification, clustering and outlier detection, and association rule discovery. Application of typical data-mining algorithms such as K-nearest neighbour or clustering could be found.

3.1 Disadvantages:

Anomaly Detection accuracy is too low.

4. PROPOSED SYSTEM

This system proposed a feasible multi-order Markov chain based framework for anomaly detection. In this approach, both the high-order Markov chain and multivariate time series are adopted to compose a scheme described in algorithms along with the training procedure in the form of statistical learning framework. To curb time and space complexity, the algorithms are designed and implemented with non-zero value table and logarithm values in initial and transition matrices. For validation, the series of system calls and the corresponding return values are extracted from classic Defense Advanced Research Projects Agency (DARPA) intrusion detection evaluation data set to form a two-dimensional test input set.

4.1 Advantages:

Anomaly Detection accuracy is high.

5. ARCHITECTURE

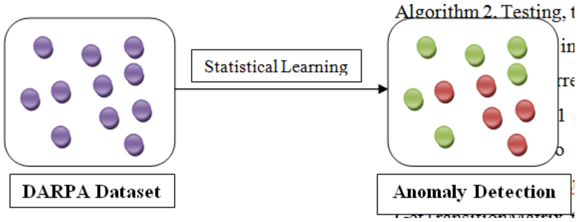


Fig 5.1 System Design

6. IMPLEMENTATION

6.1 Load Dataset:

This module loads DARPA Intrusion Detection Dataset. This dataset has n number of training sequences. Followed by, this proposed work applies training and testing for anomaly detection.

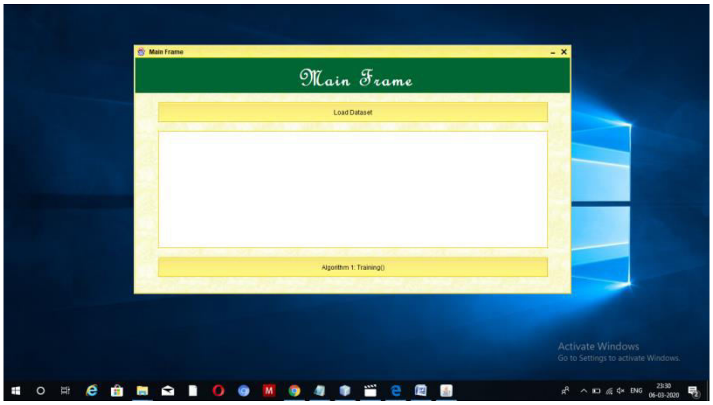
6.2 Training

Algorithm 1, Training, represents the training stage which processes the training set and generates the initial probability distribution and transition probability distribution matrices, of which the latter one is the major result of the training stage.

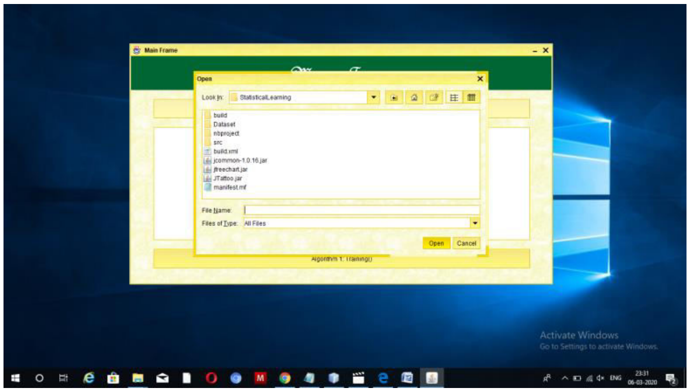
6.3 Testing

Algorithm 2, Testing, takes the two matrices and the test set in and calculates its probability of occurrence given the model trained. Algorithm 1 and Algorithm 2 also rely on two functions i.e. Increase Transition Matrix and Get Transition Matrix to calculate transition probabilities and retrieve values in these matrices.

7. OUTPUT RESULTS



Load Data set

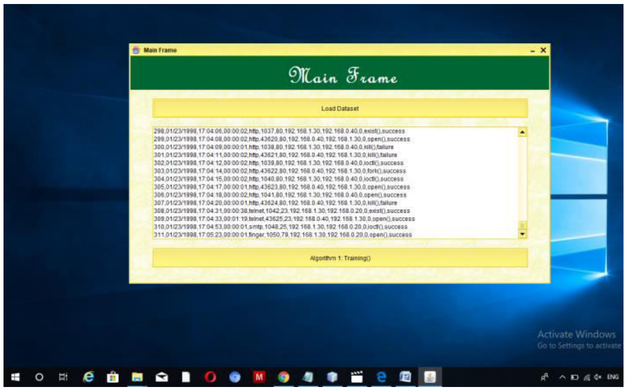


Look In: StatisticalLearning

build
Dataset
nbproject
src
build.xml
jcommon-1.0.16.jar
jfreechart.jar
JTattoo.jar
manifest.mf

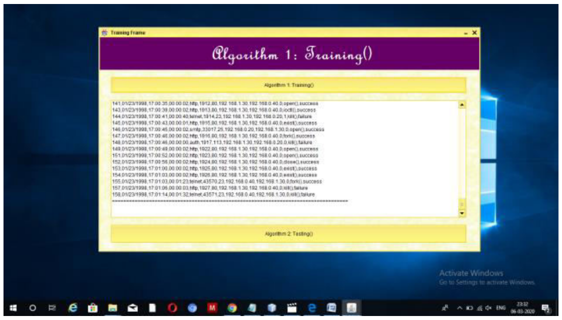
File Name:
Files of Type: All Files

Open Data set



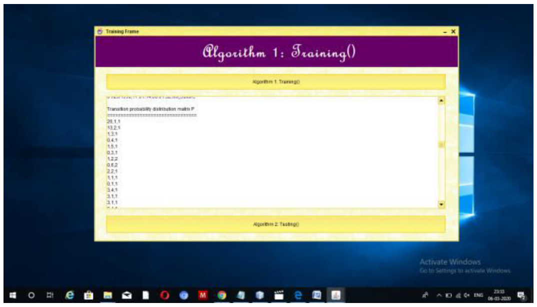
```
298,01/23/1998,17:04:06,00:00:02,http,1037,80,192.168.1.30,192.168.0.40,0,exist(),success
299,01/23/1998,17:04:08,00:00:02,http,43620,80,192.168.0.40,192.168.1.30,0,open(),success
300,01/23/1998,17:04:09,00:00:01,http,1038,80,192.168.1.30,192.168.0.40,0,kill(),failure
301,01/23/1998,17:04:11,00:00:02,http,43621,80,192.168.0.40,192.168.1.30,0,kill(),failure
302,01/23/1998,17:04:12,00:00:02,http,1039,80,192.168.1.30,192.168.0.40,0,ioctl(),success
303,01/23/1998,17:04:14,00:00:02,http,43622,80,192.168.0.40,192.168.1.30,0,fork(),success
304,01/23/1998,17:04:15,00:00:02,http,1040,80,192.168.1.30,192.168.0.40,0,ioctl(),success
305,01/23/1998,17:04:17,00:00:01,http,43623,80,192.168.0.40,192.168.1.30,0,open(),success
306,01/23/1998,17:04:18,00:00:02,http,1041,80,192.168.1.30,192.168.0.40,0,open(),success
307,01/23/1998,17:04:20,00:00:01,http,43624,80,192.168.0.40,192.168.1.30,0,kill(),failure
308,01/23/1998,17:04:31,00:00:38,telnet,1042,23,192.168.1.30,192.168.0.20,0,exist(),success
309,01/23/1998,17:04:33,00:01:19,telnet,43625,23,192.168.0.40,192.168.1.30,0,open(),success
310,01/23/1998,17:04:53,00:00:01,smtp,1048,25,192.168.1.30,192.168.0.20,0,ioctl(),success
311,01/23/1998,17:05:23,00:00:01,finger,1050,79,192.168.1.30,192.168.0.20,0,open(),success
```

Show data set screen



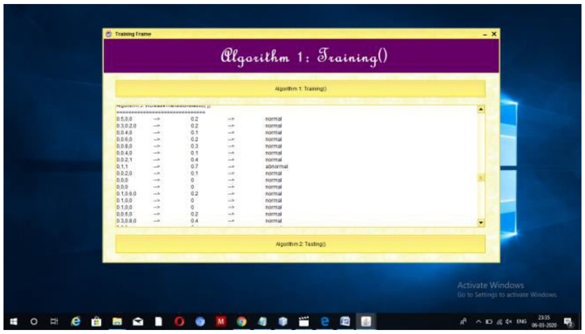
```
141,01/23/1998,17:00:35,00:00:02,http,1912,80,192.168.1.30,192.168.0.40,0,open(),success
143,01/23/1998,17:00:39,00:00:02,http,1913,80,192.168.1.30,192.168.0.40,0,ioctl(),success
144,01/23/1998,17:00:41,00:00:40,telnet,1914,23,192.168.1.30,192.168.0.20,1,kill(),failure
145,01/23/1998,17:00:43,00:00:01,http,1915,80,192.168.1.30,192.168.0.40,0,exist(),success
146,01/23/1998,17:00:45,00:00:02,smtp,33017,25,192.168.0.20,192.168.1.30,0,open(),success
147,01/23/1998,17:00:46,00:00:02,http,1916,80,192.168.1.30,192.168.0.40,0,fork(),success
148,01/23/1998,17:00:46,00:00:00,auth,1917,113,192.168.1.30,192.168.0.20,0,kill(),failure
149,01/23/1998,17:00:49,00:00:02,http,1922,80,192.168.1.30,192.168.0.40,0,open(),success
151,01/23/1998,17:00:52,00:00:02,http,1923,80,192.168.1.30,192.168.0.40,0,open(),success
152,01/23/1998,17:00:56,00:00:02,http,1924,80,192.168.1.30,192.168.0.40,0,close(),success
153,01/23/1998,17:01:00,00:00:02,http,1925,80,192.168.1.30,192.168.0.40,0,exist(),success
154,01/23/1998,17:01:03,00:00:02,http,1926,80,192.168.1.30,192.168.0.40,0,exist(),success
155,01/23/1998,17:01:03,00:01:23,telnet,43570,23,192.168.0.40,192.168.1.30,0,fork(),success
157,01/23/1998,17:01:06,00:00:03,http,1927,80,192.168.1.30,192.168.0.40,0,kill(),failure
158,01/23/1998,17:01:14,00:01:32,telnet,43571,23,192.168.0.40,192.168.1.30,0,kill(),failure
==================================================================================
```

Training Algorithm



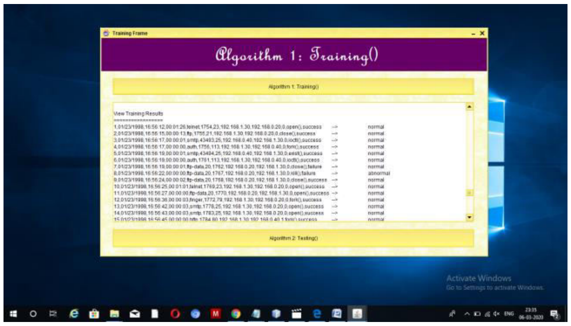
```
Transition probability distribution matrix P
===================================
28,1,1
13,2,1
1,3,1
0,4,1
1,5,1
0,3,1
1,2,2
0,6,2
2,2,1
1,1,1
0,1,1
3,4,1
3,1,1
3,1,1
```

Transition probability distribution matrix p



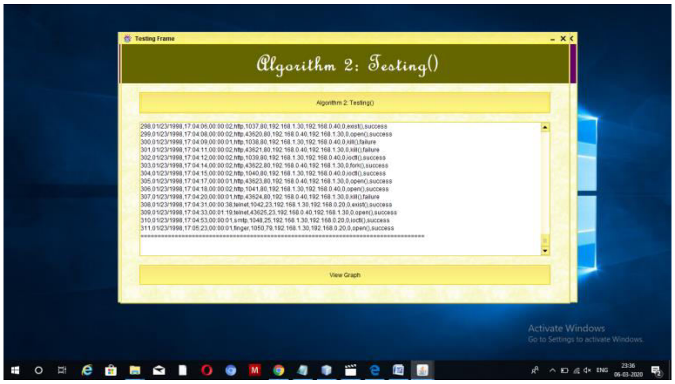
```
==============================
0,5,0,0        -->        0.2        -->        normal
0,3,0,2,0      -->        0.2        -->        normal
0,0,4,0        -->        0.1        -->        normal
0,0,6,0        -->        0.2        -->        normal
0,0,8,0        -->        0.3        -->        normal
0,0,4,0        -->        0.1        -->        normal
0,0,2,1        -->        0.4        -->        normal
0,1,1          -->        0.7        -->        abnormal
0,0,2,0        -->        0.1        -->        normal
0,0,0          -->        0          -->        normal
0,0,0          -->        0          -->        normal
0,1,0,6,0      -->        0.2        -->        normal
0,1,0,0        -->        0          -->        normal
0,1,0,0        -->        0          -->        normal
0,0,6,0        -->        0.2        -->        normal
0,3,0,8,0      -->        0.4        -->        normal
```

Algorithm -3



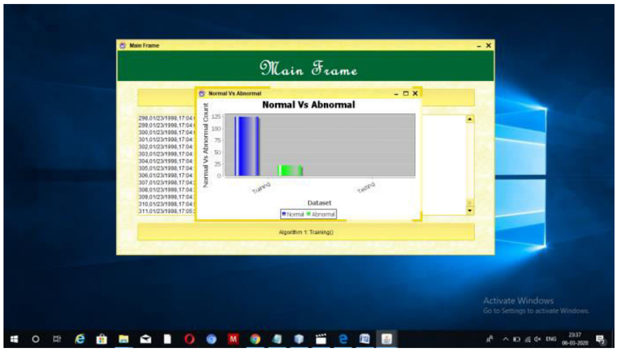
```
View Training Results
==================
1,01/23/1998,16:56:12,00:01:26,telnet,1754,23,192.168.1.30,192.168.0.20,0,open(),success      -->      normal
2,01/23/1998,16:56:15,00:00:13,ftp,1755,21,192.168.1.30,192.168.0.20,0,close(),success        -->      normal
3,01/23/1998,16:56:17,00:00:01,smtp,43493,25,192.168.0.40,192.168.1.30,0,ioctl(),success      -->      normal
4,01/23/1998,16:56:17,00:00:00,auth,1756,113,192.168.1.30,192.168.0.40,0,fork(),success       -->      normal
5,01/23/1998,16:56:19,00:00:01,smtp,43494,25,192.168.0.40,192.168.1.30,0,exist(),success      -->      normal
6,01/23/1998,16:56:19,00:00:00,auth,1761,113,192.168.1.30,192.168.0.40,0,ioctl(),success      -->      normal
7,01/23/1998,16:56:19,00:00:01,ftp-data,20,1762,192.168.0.20,192.168.1.30,0,close(),failure   -->      normal
8,01/23/1998,16:56:22,00:00:00,ftp-data,20,1767,192.168.0.20,192.168.1.30,0,kill(),failure    -->      abnormal
9,01/23/1998,16:56:24,00:00:02,ftp-data,20,1768,192.168.0.20,192.168.1.30,0,close(),success  -->      normal
10,01/23/1998,16:56:25,00:01:01,telnet,1769,23,192.168.1.30,192.168.0.20,0,open(),success     -->      normal
11,01/23/1998,16:56:27,00:00:00,ftp-data,20,1770,192.168.0.20,192.168.1.30,0,open(),success  -->      normal
12,01/23/1998,16:56:36,00:00:03,finger,1772,79,192.168.1.30,192.168.0.20,0,fork(),success     -->      normal
13,01/23/1998,16:56:42,00:00:03,smtp,1778,25,192.168.1.30,192.168.0.20,0,open(),success       -->      normal
14,01/23/1998,16:56:43,00:00:03,smtp,1783,25,192.168.1.30,192.168.0.20,0,open(),success       -->      normal
15,01/23/1998,16:56:45,00:00:00,http,1784,80,192.168.1.30,192.168.0.40,1,fork(),success       -->      normal
```

View Training results.



```
298,01/23/1998,17:04:06,00:00:02,http,1037,80,192.168.1.30,192.168.0.40,0,exist(),success
299,01/23/1998,17:04:08,00:00:02,http,43620,80,192.168.0.40,192.168.1.30,0,open(),success
300,01/23/1998,17:04:09,00:00:01,http,1038,80,192.168.1.30,192.168.0.40,0,kill(),failure
301,01/23/1998,17:04:11,00:00:02,http,43621,80,192.168.0.40,192.168.1.30,0,kill(),failure
302,01/23/1998,17:04:12,00:00:02,http,1039,80,192.168.1.30,192.168.0.40,0,ioctl(),success
303,01/23/1998,17:04:14,00:00:02,http,43622,80,192.168.0.40,192.168.1.30,0,fork(),success
304,01/23/1998,17:04:15,00:00:02,http,1040,80,192.168.1.30,192.168.0.40,0,ioctl(),success
305,01/23/1998,17:04:17,00:00:01,http,43623,80,192.168.0.40,192.168.1.30,0,open(),success
306,01/23/1998,17:04:18,00:00:02,http,1041,80,192.168.1.30,192.168.0.40,0,open(),success
307,01/23/1998,17:04:20,00:00:01,http,43624,80,192.168.0.40,192.168.1.30,0,kill(),failure
308,01/23/1998,17:04:31,00:00:38,telnet,1042,23,192.168.1.30,192.168.0.20,0,exist(),success
309,01/23/1998,17:04:33,00:01:19,telnet,43625,23,192.168.0.40,192.168.1.30,0,open(),success
310,01/23/1998,17:04:53,00:00:01,smtp,1048,25,192.168.1.30,192.168.0.20,0,ioctl(),success
311,01/23/1998,17:05:23,00:00:01,finger,1050,79,192.168.1.30,192.168.0.20,0,open(),success
=====================================================================================
```

Testing Algorithm



```
298,01/23/1998,17:04
299,01/23/1998,17:04
300,01/23/1998,17:04
301,01/23/1998,17:04
302,01/23/1998,17:04
303,01/23/1998,17:04
304,01/23/1998,17:04
305,01/23/1998,17:04
306,01/23/1998,17:04
307,01/23/1998,17:04
308,01/23/1998,17:04
309,01/23/1998,17:04
310,01/23/1998,17:04
311,01/23/1998,17:05
```

View Graph.

8. CONCLUSION

This work proposed a multi-order Markov chain based anomaly detection framework. By monitoring the relative relations between results from the different-order models, we provide a new effective indicator of anomalies. In general, due to the regular and periodical behaviors of cloud server systems, if the probability of test set given the lower-order model exceeds that given the higher-order one, it is implied that unusual events might have occurred in the system and further attentions or actions would be necessary. Besides, combining multi-dimensional inter-related sequences as a multivariate one into a single model would be another feasible approach to improve the sensitivity of detection. As shown previously, the return value series can be a useful complement to the system call series used the traditional practice. In addition, with both time and space efficiency of the Training and Testing algorithm, this approach minimizes the possibility of becoming the source of anomalies itself and is fully capable of online (or real-time) detection. The time consumption of the training stage takes no more than 15 seconds for a training set as large as 1.6 million, and for models up to the third-order combined.

9. FUTURE ENHANCEMENT

To further improve efficiency, various ways such as equivalent space construction by including an artificial state, or binary representation for sparse matrix could significantly mitigate the space complexity problem.

10. BIBLIOGRAPHY

1. Improving product marketing by predicting early reviewers on E-Commerce websites
S. Kodati, M. Dhasaratham, V. V. S. S. Srikanth, and K. M. Reddy, “Improving product marketing by predicting early reviewers on E-Commerce websites,” Deleted Journal, no. 43, pp. 17–25, Apr. 2024, doi: 10.55529/ijrise.43.17.25.

2. Kodati, Dr Sarangam, et al. “Classification of SARS Cov-2 and Non-SARS Cov-2 Pneumonia Using CNN.” Journal of Prevention, Diagnosis and Management of Human Diseases (JPDMHD) 2799-1202, vol. 3, no. 06, 23 Nov. 2023, pp. 32–40, journal.hmjournals.com/index.php/JPDMHD/article/view/3406/2798, https://doi.org/10.55529/jpdmhd.36.32.40. Accessed 2 May 2024.

3. V. Srikanth, “CHRONIC KIDNEY DISEASE PREDICTION USING MACHINE LEARNING ALGORITHMS,” IJTE, pp. 106–109, Jan. 2023, [Online]. Available: http://ijte.uk/archive/2023/CHRONIC-KIDNEY-DISEASE-PREDICTION-USING-MACHINE-LEARNING-ALGORITHMS.pdf

4. V. SRIKANTH, "DETECTION OF PLAGIARISM USING ARTIFICIAL NEURAL NETWORKS," International Journal of Technology and Engineering, vol. XV, no. I, pp. 201–204, Feb. 2023, [Online]. Available: http://ijte.uk/archive/2023/DETECTION-OF-PLAGIARISM-USING-ARTIFICIAL-NEURAL-NETWORKS.pdf

5. V. SRIKANTH, "A REVIEW ON MODELING AND PREDICTING OF CYBER HACKING BREACHES," IJTE, vol. XV, no. I, pp. 300–302, Mar. 2023, [Online]. Available: http://ijte.uk/archive/2023/A-REVIEW-ON-MODELING-AND-PREDICTING-OF-CYBER-HACKING-BREACHES.pdf

6. S. Kodati, M. Dhasaratham, V. V. S. S. Srikanth, and K. M. Reddy, "Detection of fake currency using machine learning models," Deleted Journal, no. 41, pp. 31–38, Dec. 2023, doi: 10.55529/ijrise.41.31.38.

7. "Cyberspace and the Law: Cyber Security." IOK STORE, iokstore.inkofknowledge.com/product-page/cyberspace-and-the-law. Accessed 2 May 2024.

8. "Data Structures Laboratory Manual." IOK STORE, www.iokstore.inkofknowledge.com/product-page/data-structures-laboratory-manual. Accessed 2 May 2024.

9. Data Analytics Using R Programming Lab." IOK STORE, www.iokstore.inkofknowledge.com/product-page/data-analytics-using-r-programming-lab. Accessed 2 May 2024.

10. V. Srikanth, Dr. I. Reddy, and Department of Information Technology, Sreenidhi Institute of Science and Technology, Hyderabad, 501301, India, "WIRELESS SECURITY PROTOCOLS (WEP,WPA,WPA2 & WPA3)," journal-article, 2019. [Online]. Available: https://www.jetir.org/papers/JETIRDA06001.pdf

10. V. SRIKANTH, "Secured ranked keyword search over encrypted data on cloud," IJIEMR Transactions, vol. 07, no. 02, pp. 111–119, Feb. 2018, [Online]. Available: https://www.ijiemr.org/public/uploads/paper/1121_approvedpaper.pdf

11. V. SRIKANTH, "A NOVEL METHOD FOR BUG DETECTION TECHNIQUES USING INSTANCE SELECTION AND FEATURE SELECTION," IJIEMR Transactions, vol. 06, no. 12, pp. 337–344, Dec. 2017, [Online]. Available: https://www.ijiemr.org/public/uploads/paper/976_approvedpaper.pdf

12 . SRIKANTH MCA, MTECH, MBA, "ANALYZING THE TWEETS AND DETECT TRAFFIC FROM TWITTER ANALYSIS," Feb. 2017. [Online]. Available: http://ijmtarc.in/Papers/Current%20Papers/IJMTARC-170309.pdf

14 Srikanth, V. 2018. "Secret Sharing Algorithm Implementation on Single to Multi Cloud." International Journal of Research 5 (01): 1036–41. https://journals.pen2print.org/index.php/ijr/article/view/11641/11021.

5. K. Meenendranath Reddy, et al. Design and Implementation of Robotic Arm for Pick and Place by Using Bluetooth Technology. No. 34, 16 June 2023, pp. 16–

21, https://doi.org/10.55529/jeet.34.16.21. Accessed 20 Aug. 2023.

16. Babu, Dr P. Sankar, et al. “Intelligents Traffic Light Controller for Ambulance.” Journal of Image Processing and Intelligent Remote Sensing(JIPIRS) ISSN 2815-0953, vol. 3, no. 04, 19 July 2023, pp. 19–26, journal.hmjournals.com/index.php/JIPIRS/article/view/2425/2316, https://doi.org/10.55529/jipirs.34.19.26. Accessed 24 Aug. 2023.

17. S. Maddilety, et al. “Grid Synchronization Failure Detection on Sensing the Frequency and Voltage beyond the Ranges.” Journal of Energy Engineering and Thermodynamics, no. 35, 4 Aug. 2023, pp. 1–7, https://doi.org/10.55529/jeet.35.1.7. Accessed 2 May 2024.

18. K. Meenendranath Reddy, et al. Design and Implementation of Robotic Arm for Pick and Place by Using Bluetooth Technology. No. 34, 16 June 2023, pp. 16–21, https://doi.org/10.55529/jeet.34.16.21. Accessed 20 Aug