JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 25 No. 4, pp. 105-112, April 2020
https://doi.org/10.9708/jksci.2020.25.04.105

CLIAM: Cloud Infrastructure Abnormal Monitoring using Machine Learning

Sang-Yong Choi*

*Professor, Dept. of Cyber Security, Yeungnam University College, Daegu, Korea

[Abstract]

In the fourth industrial revolution represented by hyper-connected and intelligence, cloud computing is drawing attention as a technology to realize big data and artificial intelligence technologies. The proliferation of cloud computing has also increased the number of threats. In this paper, we propose one way to effectively monitor to the resources assigned to clients by the IaaS service provider. The method we propose in this paper is to model the use of resources allocated to cloud systems using ARIMA algorithm, and it identifies abnormal situations through the use and trend analysis. Through experiments, we have verified that the client service provider can effectively monitor using the proposed method within the minimum amount of access to the client systems.

▸ **Key words**: Cloud computing, ARIMA algorithm, Machine learning, Resource, Monitoring

[요 약]

초연결, 지능화로 대표되는 4차 산업혁명에서 클라우드컴퓨팅은 빅데이터와 인공지능 기술을 실현하기 위한 기술로 주목받고 있다. 클라우드컴퓨팅이 확산됨에 따라 이에 대한 다양한 위협 또한 증가하고 있다. 클라우드컴퓨팅 환경의 위협에 대응하기 위한 하나의 방법으로 본 논문에서는 IaaS 서비스 제공자가 클라이언트에게 할당한 자원에 대해 효과적인 모니터링 할 수 있는 방법을 제안한다. 본 논문에서 제안하는 방법은 할당된 클라우드 자원의 사용량을 ARIMA 알고리즘으로 모델링 하고, 평시 사용량과 추이 분석을 통해 비정상 상황을 식별할 수 있는 방법이다. 본 논문에서는 실험을 통해 제안한 방법을 이용하여 클라우드 서비스 제공자가 클라이언트 시스템에 대한 최소한의 권한으로 효과적으로 모니터링 할 수 있음을 보였다.

▸ **주제어**: 클라우드컴퓨팅, 아리마 알고리즘 머신러닝, 리소스, 모니터링

• First Author: Sang-Yong Choi, Corresponding Author: Sang-Yong Choi
*Sang-Yong Choi (spikechoi@ync.ac.kr), Dept. of Cyber Security, Yeungnam University College
• Received: 2020. 03. 25, Revised: 2020. 04. 09, Accepted: 2020. 04. 09.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

사물인터넷, 빅데이터, 인공지능 등으로 대표되는 4차 산업혁명 시대가 도래하고 있다. 조직에서 생산하거나 보유하는 데이터의 양이 예측할 수 없을 정도로 방대해지고, 데이터의 형태 또한 텍스트, 데이터, 영상 등을 포함하는 다양한 포맷으로 증가하고 있다. 이와 같은 데이터는 산업 분야의 가치와 비즈니스의 가치에 있어 더욱 중요해지고 있다[1]. 특히, 많은 양의 데이터를 효과적으로 처리하여 데이터로부터 가치 있는 정보를 획득하기 위해 컴퓨팅 성능이 중요해지고 있다. 이처럼, 초연결(Hyper-Connectivity), 지능화(Intelligence)로 대표되는 4차 산업혁명에는 빅데이터, 클라우드컴퓨팅(이하, “클라우드”로 표기), 인공지능 등의 기술이 핵심이 되고, 이 중 특히 클라우드 기술은 빅데이터와 인공지능 기술을 실현하기 위한 기반으로 핵심역할을 담당하고 있다[2].

기업에서는 이러한 기능을 실현하기 위해 아마존의 AWS(Amazon Web Services)[3], 마이크로소프트의 AZURE[4] 등과 같은 전문 클라우드 서비스를 활용하거나 자체적으로 클라우드 시스템을 구축하여 활용하고 있다. 하지만, 클라우드 서비스의 활용이 증가함에 따라 가상화 취약점, 정보위탁에 따른 정보유출 위험, 자원공유 및 집중화에 따른 서비스 장애, 단말 다양성에 따른 정보유출, 분산처리에 따른 보안 적용의 어려움, 법 및 규제의 측면 등 클라우드 환경의 특성에 따른 보안 위협 또한 증가하고 있다[5-6].

이와 같은 다양한 위협에 대응하기 위해 다양한 형태의 클라우드 보안서비스 또는 오픈소스를 활용한 클라우드 보안서비스가 등장하고 있으나, 더욱 안전한 클라우드 환경의 운영을 위해서는 클라우드 환경으로 유입되는 이상행위의 식별과 관리가 무엇보다 중요하다. 하지만 클라우드 환경의 특성상 고객에게 할당한 클라우드 환경에 포함되는 데이터에 대한 안전한 관리 이슈로 인해 클라우드 인프라의 관리자라 하더라도 개별 환경에 대한 모니터링은 제한적으로 이루어질 수밖에 없다. 이러한 이유로, 일반적으로 기존 시스템에서 사용하는 침입탐지시스템으로 클라우드 환경 전체에 대한 완전한 모니터링을 수행하는 데는 한계가 있는 것이 현실이다. 본 논문에서는 이와 같은 문제점에 대한 한 가지 해결방안으로 머신러닝을 이용하여 클라우드 환경을 모니터링을 하는 방법(CLIAM:Cloud Infrastructure Abnormal Monitoring using Machine Learning)을 제안한다. 제안하는 방법은 머신러닝의 다양한 알고리즘 중 시계열 분석이 가능한 ARIMA 알고리즘을 이용하여 리소스 사용에 대한 모델을 만들고, 이 모델을 활용하여 할당된 개별 클라우드 인프라(이하 “클라우드 클라이언트”로 표기)에 대한 세부적인 접근이 없이도, 인프라 관리자가 접근할 수 있는 수준에서 클라우드 환경의 이상 증상을 식별할 수 있게 한다. 즉, 클라우드 클라이언트 리소스 사용량과 클라우드 전체 인프라의 사용량 관계에서 비정상적인 상황을 식별하여 관리자가 비정상적인 상황을 빨리 인지하고 클라우드 클라이언트 담당자에게 알려줄 수 있게 하여 궁극적으로 클라우드 환경에 대한 위협뿐만 아니라 과다한 자원 소모를 방지하여 효율적인 자원관리가 가능하도록 한다.

본 논문은 2장에서 클라우드 기술과 클라우드 환경의 위협 그리고 자원 사용량 분석을 위한 머신러닝 알고리즘 등을 설명한다. 그리고 3장에서 클라우드 환경의 모니터링 아키텍처를 제안하며, 4장에서 실험을 통해 제안하는 방법이 효과가 있음을 검증한다.

II. Related Work

1. Concepts of cloud computing

2017년 7월부터 시행되고 있는 “클라우드컴퓨팅 발전 및 이용자 보호에 관한 법률” 제2조 제1항에서 클라우드 컴퓨팅은 “집적, 공유된 정보통신기기, 정보통신 설비, 소프트웨어 등 정보통신자원을 이용자의 요구나 수요 변화에 따라 정보통신망을 통하여 신축적으로 이용할 수 있도록 하는 정보처리 체계”라고 설명하고 있다.

위키백과에는 클라우드를 “클라우드(인터넷)를 통해 가상화된 컴퓨터의 시스템 리소스를 요구하는 즉시 제공하는 것이라고 설명하고 있다. 더불어 클라우드의 특징으로 조직을 위한 민첩성의 개선, 비용 절감, 장치 및 위치 독립성, 유지보수 용이성 등을 장점으로 말하고 있다. 이와 같은 정의를 종합한 클라우드의 개념은 Fig.1과 같다[7]

클라우드 서비스 모델은 전통적으로 응용프로그램 등 소프트웨어를 제공하는 서비스인 SaaS(Software as a Service), 응용프로그램 등 소프트웨어의 개발, 배포, 운영, 관리 등을 위한 환경을 제공하는 PaaS(Platform as a Service), VM-Ware, AWS 등으로 대표되는 서버, 저장장치, 네트워크 등을 제공하는 IaaS(Infrastructure as a Service) 등으로 구분한다[8].

본 논문에서의 관점은 IaaS이다. 본 논문에서는 IaaS를 제공하는 조직에서 제공한 인프라가 잘 활용되고 있는지, 제공한 인프라에서 비정상적인 증상이 발생하지는 않는지 등을 효과적으로 모니터링 할 수 있는 방법을 제안한다.

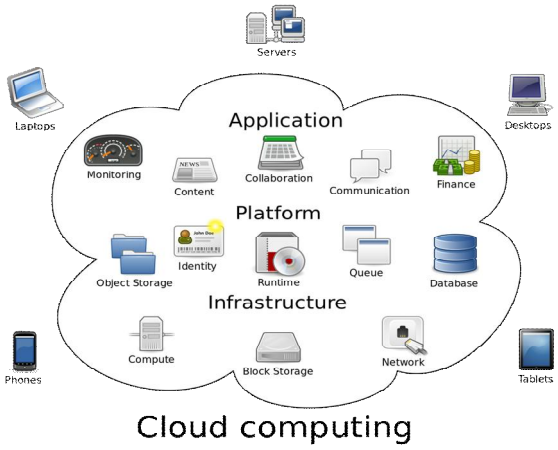

Fig. 1. Cloud Computing[7]

2. Security threats for cloud computing

CSA(Cloud Security Alliance)에서 2017년 클라우드에 관한 12가지 위협을 발표하였다. 동 보고서에는 클라우드 위협을 기술적인 위협, 내부자에 의한 위협 등으로 분류한다. 주요 위협으로 데이터 유출, 불충분한 ID 및 자격증명 그리고 접근관리, 안전하지 않은 API와 인터페이스, 시스템 취약점, 계정도용, 악의적인 내부자, APT(Advanced Persistent Threats), 데이터 손실, 불충분한 실사, 클라우드 서비스 남용과 악의적인 사용, DoS(Denial of Service), 공유기술 취약점, 스펙트라와 멜트다운 등이 있다.

이와 같은 위협 대부분은 기술적인 취약점으로 발생하며, 침입탐지 시스템, 침입방지시스템 등과 같은 보안시스템으로 감지하거나 탐지할 수 있을 것으로 생각된다. 다만, 안전하지 않은 API, 시스템 취약점, 계정도용, 악의적인 내부자, 서비스 남용 등과 같은 위협은 내부와 외부구간의 접점에 설치되는 침입탐지시스템이나, 내부 세그먼트 사이에 구성되는 침입탐지시스템, 즉 경계 구간에 설치하는 시스템으로 감지하기가 어렵다[10].

따라서 클라우드 서비스에 대한 위협과 오남용을 식별하기 위한 효과적인 방법이 필요하다.

3. Machine leaning and ARIMA algorithm

머신러닝은 인공지능의 한 분야이며, 학습 알고리즘을 이용하여 다양한 상황을 컴퓨터가 학습하는 방법을 연구하는 분야이다[11]. 머신러닝을 위한 알고리즘으로는 지도학습(Supervised Learning), 비지도 학습(Non-Supervised Learning), 준 지도 학습(Semi-supervised learning), 강화학습(Reinforcement learning) 등이 있다[12].

지도학습은 훈련 데이터로부터 하나의 함수를 유추해내기 위한 머신러닝의 방법이다. 즉, 훈련 데이터로부터 함수가 도출되면 도출된 함수를 최적화하고 이를 활용하여 새로운 데이터 또는 테스트 데이터를 분류하는 접근법이다. 비지도 학습은 지도학습과는 달리 입력 값에 대한 목표치가 주어지지 않고 자율적으로 특성을 학습한다. 준 지도학습은 범주로 목표 값이 표시된 데이터와 표시되지 않은 데이터를 모두 훈련에 사용하여 지도학습과 비지도 학습의 장점을 모두 차용하는 방법이다. 강화학습은 어떤 상태에서 학습 에이전트가 현재 상태를 인식한 후 선택 가능한 행동 중 보상함수에 의해 보상이 최대로 주어지는 행동 또는 순서를 예측하는 학습방법이다.

즉, 머신러닝은 공통으로 주어진 데이터를 컴퓨터에 학습시켜 학습된 데이터로부터 정상과 비정상 분류, 비정상 행위 식별 등 새로운 데이터에 대한 분류와 분석을 수행하는 방법이다. 그리고 이를 통해 앞으로 일어날 일 등을 예측할 수 있도록 접근하는 연구 분야이다[12]. 학습 알고리즘 중 ARIMA(Auto regressive Integrated Moving Average) 알고리즘은 과거의 데이터를 사용하여 과거의 데이터가 가지고 있는 추세(Momentum)를 반영하여 상황을 예측하는 시계열 분석 알고리즘이다[13]. ARIMA 모델은 시간에 따라 변하는 데이터를 활용하여 미래 상태를 예측하는 데 주로 활용된다. 본 논문에서 자원 사용량의 변화를 분석하기 위해 ARIMA 모델을 사용한다.

4. Limitations and improvements

살펴본 바와 같이 클라우드 환경의 보안 위협은 기술적인 위협과 관리적인 위협(인적분야)이 있으며, 기술적인 위협 중 많은 부분이 현재의 모니터링 시스템으로 위협과 이상 증상을 판단하기에 한계가 있다. 특히 클라우드 환경 중 IaaS의 경우에는 인프라 관리하는 서비스제공자는 인프라를 클라이언트에게 제공하지만, 인프라 내에 포함된 데이터나 애플리케이션에 대한 접근 권한이 부여되지 않으며, 비록 서비스제공자가 보안서비스를 제공한다 하더라도, 클라이언트가 위협을 적절히 관리하지 못한다면 동일 인프라 내에서 사이버 위협이 발생할 수 있고, 발생한 위협은 동일 인프라를 활용하는 제3의 클라이언트에게 영향을 줄 수 있다. 하지만 인프라 관리자는 클라이언트에 제공한 인프라에 대한 접근 권한이 존재하지 않기 때문에 문제를 인식할 수 없을 가능성이 커진다.

이와 같은 문제에 접근하려는 방법으로 인프라 제공자 범위에서 관리 가능한 데이터를 적극적으로 활용하여 클라우드 클라이언트 측의 위협과 이상징후를 식별하고 분석할 수 있어야 한다. 관리자가 접근할 수 있고 관리 가능한 데이터로는 효과적인 IaaS 서비스를 제공하기 위한 목적으로

각 클라이언트에게 할당된 자원(CPU, Memory, Disk 등)의 사용상태 정보에 접근할 수가 있으며, 이 정보를 활용하여 시계열 머신러닝의 접근방법으로 이상 분석을 수행할 수 있을 것이다. 자원의 사용량 모니터링과 예측을 위한 알고리즘으로는 머신러닝 알고리즘 중 ARIMA 모형이 가장 효과적일 것으로 판단된다. ARIMA 모형의 경우 시계열 분석에 적합한 알고리즘이며, 자원 사용량의 경우 수집되는 정보에서 시간 데이터가 포함되기 때문이다.

따라서 ARIMA 알고리즘을 이용하여 개별 클라이언트와 전체 클라우드 인프라의 자원 사용량을 분석하여 이상 징후를 분석하는 방법으로 이 문제를 해결하기 위한 접근방법을 찾을 수 있을 것으로 예상한다.

III. CLIAM: Cloud Infrastructure Abnormal Monitoring using Machine Learning

1. Assumptions and Prerequisites

효과적인 클라우드 인프라 모니터링을 위한 방법으로 활용하기 위해 클라우드 환경의 특성을 살펴볼 필요가 있다. 먼저 가상머신으로 대표되는 클라우드 인프라는 호스트 시스템에서 가상머신을 각 클라이언트에 할당하고 클라이언트가 사용할 리소스를 정의하게 된다. 클라이언트에서 활용 가능한 리소스는 동적 또는 정적으로 할당할 수 있다 [14-15]. 그리고 대용량의 클라우드 인프라를 제공하는 환경에서는 단일 클라이언트의 리소스 사용량이 전체 클라우드 인프라에 미치는 영향이 크지 않게 된다. 즉, 하나의 클라우드 클라이언트에서의 자원 사용량의 변화는 전체적인 자원 사용량의 측면에서는 무시할 만큼 작은 변화일 수 있다.

클라우드 인프라 관리자가 개별 클라우드 환경에 접속하여 내부데이터에 접근할 수 있게 되면 기밀성과 무결성을 보장할 수 없다. 따라서, 인프라 관리자는 개별 클라우드 환경에 접속할 권한이 없고, 개별 클라우드 환경에서 수집할 수 있는 최소한의 정보(CPU, Memory, Disk, Network에 대한 사용량) 만에 접근할 수 있다. 하지만 인프라 관리자는 개별 클라우드의 이상 상태를 알 수 있고, 예측할 수 있어야 한다.

개별 클라우드 클라이언트의 입장에서는 악성코드에 감염되거나, 애플리케이션이 이상 동작을 할 때는 평소 대비 리소스 사용량의 변화가 크게 나타난다. 이는 평소 사용자의 사용이 많은 시간대의 사용량 추이와 별개로 나타나게 되며, 클라이언트 인프라 전체에 영향을 줄 수 있다.

제안하는 CLAIM은 단순한 개별 클라우드 클라이언트의 사용량의 추이 분석으로 접근을 지양하고, 개별 클라우드 클라이언트의 사용량과 전체 클라우드 인프라의 사용량과의 관계를 분석하여 이를 통해 이상을 식별한다. 실제 분석하는 알고리즘은 다음 절에서 설명한다.

2. Architecture and algorithm

CLIAM의 전체적인 구성은 개별 클라우드 시스템으로부터 수집된 데이터를 분석할 수 있도록 정규화하는 모듈과 정규화가 완료된 데이터를 ARIMA 기반 분석기에서 분석할 수 있도록 데이터모델을 만드는 모델러, 그리고 알고리즘에 의해 이상 상황을 판단하는 분석기로 구성된다. Fig. 2에서 수집 부분은 CLIAM 시스템에 포함하지 않았다. 이유는 리소스를 수집하는 방법이 다양하며 어떠한 시스템에서도 활용할 수 있는 표준화된 수집방법을 지원하게 하기 위함이다.

정규화 모듈은 수집된 데이터를 정규화하기 위해 사용된다. 정규화는 모든 수집된 데이터에 대해 식(1)과 같은 형태로 정규화한다. 즉, CPU, Memory, Disk, Network 각각의 사용량에 대해 수집된 시간을 기준으로 데이터를 정규화한다. 이렇게 함으로 단위 시간별 사용량의 추이를 분석할 수 있게 된다.

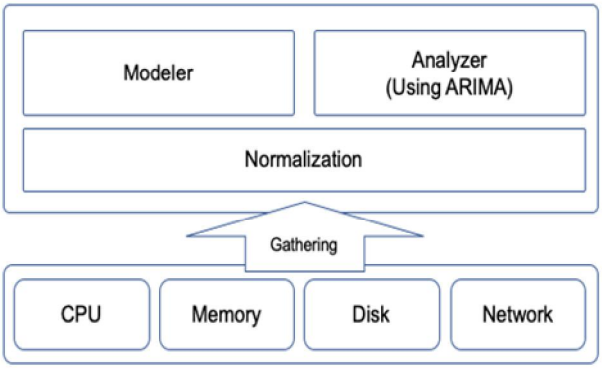

Fig. 2. System Architecture

$$
\begin{aligned}
&TimeStamp,\ CPU_1, CPU_2, CPU_3, \cdots CPU_n \quad CPU_n \in CPU\ Usage\ of\ VirtualMachine\ n \\
&TimeStamp,\ MEM_1, MEM_2, MEM_3, \cdots MEM_n \quad MEM_n \in Memory\ Usage\ of\ VirtualMachine\ n \\
&TimeStamp,\ DSK_1, DSK_2, DSK_3, \cdots DSK_n \quad DSK_n \in Disk\ Usage\ of\ VirtualMachine\ n \\
&TimeStamp,\ NT_1, NT_2, NT_3, \cdots NT_n \quad NT_n \in Ntwork\ Usage\ of\ VirtualMachine\ n
\end{aligned} \tag{1}
$$

동적 자원할당 방식을 사용하는 클라우드 환경에서 각 클라이언트에 할당된 리소스의 합은 물리적인 전체 리소스보다 클 수 있다. 즉, 클라우드 환경의 전체 메모리가 100G byte이고, 클라우드 환경에 할당된 클라이언트가 50대라고 할 때, 각 클라우드 클라이언트에 할당할 수 있는 최대 메모리는 산술적 수치인 2Gbyte를 넘어설 수 있다. 하지만, 클라우드 자원이 동적할당 방식을 사용한다면, 각 클라우드 클라이언트에 4G byte씩 할당할 수 있으며, 이 경우 현재 사용량 전체 합이 100G byte가 넘지 않으면 환경에 문제가 없다.

이를 다르게 해석하면 하나의 클라우드 클라이언트에 이슈가 발생하여 해당 클라이언트가 많은 양의 자원을 소모한다면, 정상적인 나머지 클라이언트는 약속된 자원보다 사용할 수 있는 자원이 적게 되고, 이는 전체적으로 클라우드 자체에 대한 이슈가 될 수 있다.

CLIAM의 모델러와 분석기는 이와 같은 현상, 즉 특정 클라이언트가 허용범위 이내이지만 자원을 점유하는 것을 식별한다. 식별을 위해 정의한 속성은 Table 1과 같다.

Table 1. Attrribute

Attribute	Description
$R_m VM_n U_t$	Usual day resource(CPU, Memory, Disk, Network) usage at *t* time of the *n* th virtual machine
$R_m VM_n N_t$	Current resource(CPU, Memory, Disk, Network) usage at *t* time of the *n* th virtual machine
$R_m VM_n P_t$	Resource usage at *t* time predicted by usage up to *t-1* time of the *n* virtual machine
R_m : *Resource(m : Memory, c : CPU, d : Disk)* VM_n : *Virtual Machine n,* P_t : *AVerage usage at t time* N_t : *Current usage at t time,* P_t : *Predicted usage at t time*	

모델러는 정규화된 데이터를 활용하여 분석기에서 분석이 가능하도록 각각의 속성을 학습하고 모델을 생성한다. 이때, 세 번째 속성 즉, ARIMA를 적용하여 현재 이전 시점까지의 데이터를 기반으로 현재 시점에 예상되는 리로스 사용량을 예측한다. 또한, 평소의 유사한 시점의 자원 사용량의 평균을 산출한다. 분석기는 이 2가지 학습데이터와 현재 시점에서 수집된 데이터를 기반으로 이상을 판단한다.

분석기는 두 가지의 기준으로 이상 여부를 판단한다. 첫 번째 기준은 평소 사용량 대비 현재 사용량의 차이이다. 두 번째 기준은 예측된 사용량 대비 현재 사용량의 차이이다. 이를 합하여 식으로 나타내면 식(2)와 같다.

$$2 \times R_m VM_n N_t - \frac{R_m VM_n U_t + R_m VM_n P_t}{2} \qquad (2)$$

만약 위 식(2)의 값이 0에 가깝다면 현재 시스템은 예상된 대로 평소와 유사하게 동작하고 있음을 나타낸다. 식(2)의 값이 음수라면 현재의 사용량이 평소 또는 예측 사용량보다 작다는 것을 나타낸다. 이는 사용자가 줄어들었거나, 동적 자원할당 시스템에서 타 시스템이 많은 자원을 사용하고 있어서 현재 클라이언트가 자원을 활용하지 못하고 있는 과부하의 상태를 나타낸다. 식(2)의 값이 양수라면 현재 클라이언트가 자원을 예상보다 너무 많이 소모하고 있음을 나타낸다. 이는 클라이언트의 자원 사용량이 허용범위 이내더라도 관리자 입장에서는 이러한 상황이 지속된다면 확인할 필요가 있다.

이러한 전체 과정은 Fig. 3과 같이 이루어진다. 먼저 정규화를 거친 데이터를 학습하고, 학습결과와 정규화된 데이터는 분석기의 입력으로 사용된다. 관리자에 의해 판단되고 판단이 완료된 데이터는 학습데이터로 재사용할지 제외할지를 결정한다.

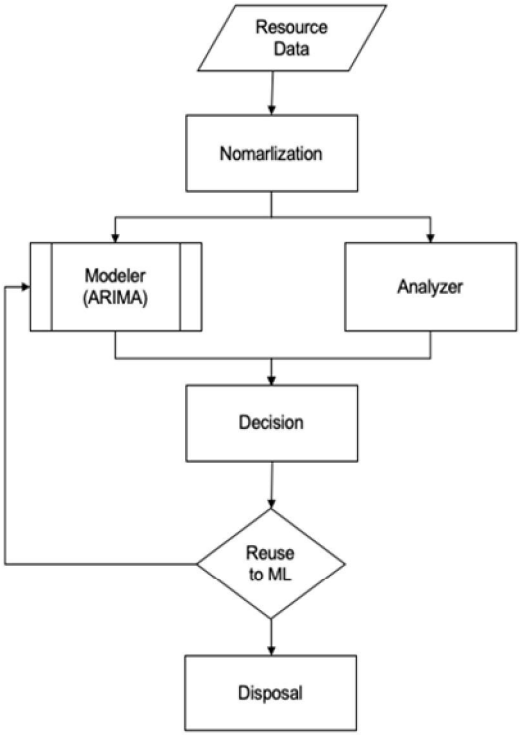

Fig. 3. Flow chart

IV. Experiment

1. Experimental environment

CLAIM의 효과를 검증하기 위한 실험환경은 Table. 2와 같다. 실험환경구성은 Hyper-V 환경에서 운영되는 실습용 가상서버를 이용하였다. 실험환경에서 가상서버의 메모리 사용량을 동적으로 설정하였다. 메모리 사용량 실험을 위해 리눅스 시스템의 부하 테스트 프로그램인 stress[16]를 이용하였다.

Table 2. Experimental Environment

System	Spec
Host	• CPU : 8CPUs * Intel(R) Xeon • Mem : 32G • Disk : 6TB
VM_1	• CPU : 1CPU • Memory : 512MB ~ 16G Dynamic • Disk : Init 20G Dynamic
VM_2	• CPU : 2CPU • Memory : 512MB ~ 4G Dynamic • Disk : Init 20G Dynamic
VM_3	• CPU : 2CPU • Memory : 512MB ~ 16G Dynamic • Disk : Init 20G Dynamic
VM_4	• CPU : 4CPU • Memory : 512MB ~ 16G Dynamic • Disk : Init 20G Dynamic

2. Experiment result

실험의 방법은 Stress 프로그램을 자동 스트립트로 하여 4개의 가상서버에서 업무시간 동안 실행되도록 만든 후 5일간의 사용량을 1분 단위로 수집한 평균 사용량을 산출하고, 수집된 데이터를 ARIMA 알고리즘을 적용하여 학습시켰다. 그리고 평소 사용패턴과 다르게 가상서버를 사용하여 현재 시스템의 리소스 사용량을 측정하였다. 측정 대상 리소스는 메모리로 선정하였으며, 나머지 리소스는 실험에서 제외하였다. 이후, 현재 수집된 데이터와 평균 데이터, ARIMA 학습데이터를 식(2)에 적용하였다.

먼저 5일간 4대의 가상머신에서 수집한 메모리 사용량의 추이는 Fig.4와 같다. 업무시간 중에 가상머신에서 사용하는 메모리 총 메모리 사용량은 약 25G byte로 측정된다. 가상머신이 운영되는 Hyper-V호스트 환경의 사용량을 제외하면 거의 물리적 메모리의 100%에 가까운 사용량을 보인다. 가상머신 자체는 업무시간 이후에는 사용하지 않기 때문에 업무시간이 종료된 후에는 사용량이 급격히 떨어지고 야간 시간 동안 최소 사용량으로 유지되는 것을 확인할 수 있다.

Fig. 4. Usual day memory usage

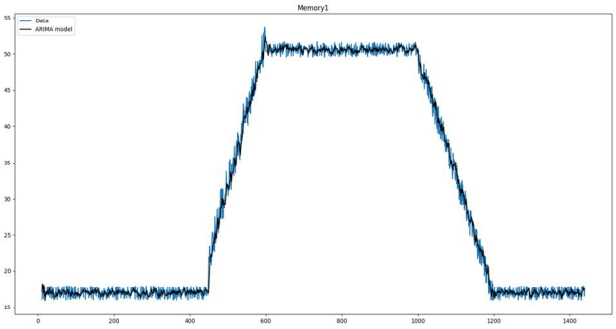

Fig. 5. ARIMA Model for VM_1

가상머신 중 VM_1의 데이터를 ARIMA 알고리즘으로 학습하여 모델을 생성하였다. 생성한 모델은 Fig.5와 같다. 자원 사용량의 변화에 따라 ARIMA 예측 모델은 관측된 데이터를 매우 잘 나타내고 있음을 알 수 있다.

그 이후 VM_1의 메모리 사용량을 높인 후 하루 동안의 사용량 변화를 Fig.6과 같다. VM_1시스템의 메모리 사용량의 증가를 위해 Stress의 옵션을 조정하여 실행한 결과 메모리 사용량이 최대 70%까지 증가함에 따라, 상대적으로 평시 사용량이 많았던 VM_4의 메모리 사용량이 상당한 수준으로 감소하였음을 확인하였다.

단순히 Fig. 6만 본다면 급격한 증가나 감소가 보이지 않기 때문에 클라우드 시스템 전체에 큰 문제가 없는 것으로 보인다. 하지만 평시 사용량과 비교해 보면 VM_1과

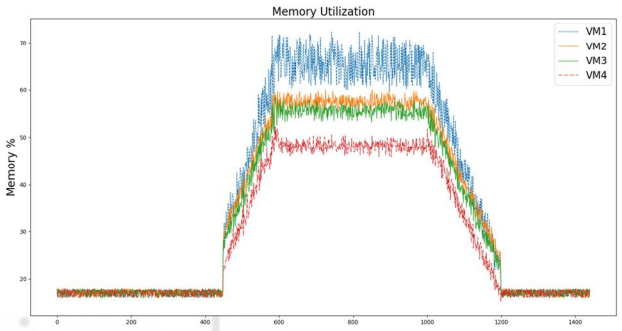

Fig. 6. Abnormally usage

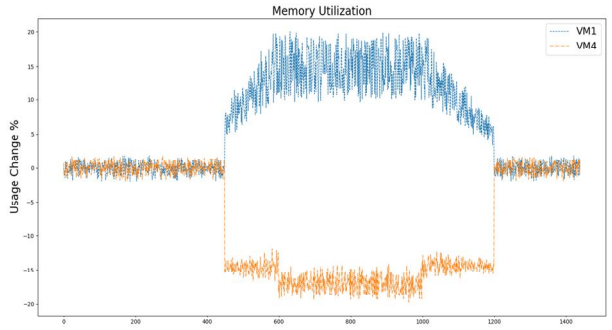

Fig. 7. Decistion Graph

VM_4가 비정상적인 상황임을 인지할 수 있다. 마지막으로 측정된 데이터를 식(2)에 적용하면 Fig. 7과 같은 결과를 확인할 수 있다. Fig.7에 따르면, 양수(+)영역을 차지하는 VM_1의 사용량은 평소 예측치 대비 너무 많은 리소스 사용량을 보이고 있으며, 음수(-)영역을 차지하는 VM_4의 경우에는 평소 대비 너무 적은 리소스 사용량을 보이고 있다.

이처럼 메모리 사용량을 대상으로 실험한 결과 제안하는 방법은 클라우드 환경의 관리자가 각각 클라이언트에게 할당한 클라우드 자원을 효율적으로 모니터링 할 수 있을 것을 알 수 있었다.

V. Conclusions

4차 산업혁명의 기술에 따라 모든 사물이 연결되는 초연결 사회가 다가오고 있으며, 이에 따라 연결된 시스템에서 발생하는 많은 양의 정보를 효과적으로 처리하기 위해 클라우드, 빅데이터와 같은 새로운 기술이 발전하고 있다. 또한, 클라우드 환경의 이상징후를 발견하기 위한 효과적인 모니터링 방법에 관한 연구가 지속되고 있으나 한계가 있는 실정이다.

본 연구에서는 IaaS 환경에서 클라우드 시스템 관리자가 제한된 권한으로 전체 클라우드 환경의 자원 사용량 모니터링을 하고, 이를 통해 이상징후를 발견할 수 있는 알고리즘을 제안하였다. 제안하는 알고리즘은 머신러닝 알고리즘 중 ARIMA 알고리즘을 이용하여 평소 클라우드 클라이언트의 자원 사용량을 모델로 만고, 이를 기반으로 현재의 사용량 변화를 분석하여 하나의 클라우드 클라이언트의 변화가 전체 클라우드 시스템에 미치는 영향을 확인할 수 있었다. CLIAM을 활용한다면, 클라우드뿐만 아니라 기존 래거시 시스템의 자원 사용량 모니터링 및 이를 통한 장애, 사이버 위협 등의 발생을 보다 효과적으로 식별할 수 있을 것으로 기대한다. 다만, 현재의 방법은 관리자가 제한된 권한 내에서 소극적으로 분석하는 방법으로 특정 시스템의 사용량 변화가 해당 클라우드 클라이언트의 이상 증상으로 인한 것인지 클라우드에 할당된 타 시스템의 영향으로 인한 것인지를 정확하기 파악하기에는 한계가 있다. 향후, 지속적으로 보다 정확도 높은 이상징후 탐지 방법에 관한 연구를 지속할 것이다.

ACKNOWLEDGEMENT

This work was supported by the Yeungnam University College Research Grants in 2019.

REFERENCES

[1] M. J. Lee, "*Internet & Information Security:Big Data and the Utilization of Public Data*", KISA, PP. 47~64, 2011

[2] hyun seok jeong, "A Study on the Application of Cloud Computing in the 4th Industrial Revolution", Korean Institute of Communications and Information Sciences, Vol. 44, No. 6, pp. 1213-1222, 2019, DOI:10.7840/kics.2019.44.6.1213

[3] Amazon AWS, *https://aws.amazon.com/ko/*

[4] Mocrosoft Azure, *https://azure.microsoft.com/ko-kr/*

[5] *Cloud Services Information Protection Guide*, KISA, pp. 1~98, 2011

[6] *Cloud Computing Issues and Status*, KEIT, pp. 1~48, 2011

[7] Cloud Computing, *https://en.wikipedia.org/wiki/Cloud_computing*

[8] Jihye Shin, "Regulations and the standard form contract for the user protection on the cloud computing service", HUFS Law Review, Vol. 40, No. 4 pp.117-140, 2016, DOI : 10.17257/hufslr.2016.40.4.117

[9] "*Top Threats to Cloud Computing Plus: Industry Insights*", Cloud Security Alliance, Dec. 2017

[10] Vyas, Giriraj, Sanjay Meena, and Pramendra Kumar. "Intrusion Detection Systems: A Modern Investigation." International Journal of Engineering, Management & Sciences (IJEMS) Vol. 1, No. 11, 2014.

[11] Alpaydin, "*Ethem. Introduction to machine learning*", MIT press, 2020.

[12] Patric R. Nicolas, "*Scala for Machine Learning*", Packt Publishing Ltd, Dec. 2015,

[13] Contreras, Javier, et al. "ARIMA models to predict next-day electricity prices." IEEE transactions on power systems, Vol. 18, No. 3, pp.1014-1020. 2003, DOI: 10.1109/TPWRS.2002.804943

[14] Yin-Goo Yim, Hyun-Wook Jin,."Analysis of Impact of Multi-core CPU Bandwidth in Docker Containers", KIISE Transactions on

Computing Practices, Vol. 24, No. 12, pp.675-680, 2018, DOI: 10.5626/ktcp.2018.24.12.675

[15] D. Merkel, "Docker: Lightweight Linux Containers for Consistent Development and Deployment", Journal of Linux Journal 2014, Vol. 2014, No. 239, pp. 2, May. 2014.

[16] STRESS, *https://www.cyberciti.biz/faq/stress-test-linux-unix-server-with-stress-ng/*

Authors

Sang-Yong Choi received his B.S. degree in Mathematics and M.S. degree in Computer Science, both from Hannam University in 2000 and 2003, and Ph.d degree in Interdisciplinary of Information Security from Chonnam National University in 2014, Dr. Choi is a assistant professor at the Dept. of Cyber Security in Yeungnam University College, Daegu, Korea. His research interests are in web security, network security and cloud computing security.