제 19 회 한국정보처리학회 춘계학술발표대회 논문집 제 10 권 제 1 호 (2003. 5)

Viterbi 알고리즘을 이용한 HMM기반 침입탐지 시스템의 침입 유형 판별

구자민°, 조성배
연세대학교 컴퓨터 과학과
e-mail : icicle@candy.yonsei.ac.kr, sbcho@cs.yonsei.ac.kr

Attack Type Discrimination for HMM-based IDS Using Viterbi Algorithm

Ja-Min Koo°, Sung-Bae Cho
Dept. of Computer Science, Yonsei University

요 약

정보통신 구조의 확산 및 기술이 발전함에 따라 전산 시스템에 대한 침입과 피해가 증가되고 있는 실정이다. 이에 비정상행위 기반 침입탐지 시스템에 대한 연구가 활발히 진행되고 있는 가운데 특히, 시스템 호출 감사자료 척도에 은닉 마르코프 모델(HMM)로 모델링 하는 연구가 많이 이루어지고 있다. 하지만, 이는 일정한 임계값 이하의 비정상행위만을 감지할 뿐, 어떠한 유형의 침입인지를 판별하지 못한다. 본 논문에서는, 이러한 침입탐지 시스템의 맹점을 보완하기 위하여 Viterbi 알고리즘을 이용하여 상태 변화를 분석한 후, 어떤 유형의 침입이 발생하였는지를 판별하는 방법을 제안하고, 실험을 통해 제안한 시스템의 가능성을 보인다.

1. 서론

정보통신 기술의 발전과 네트워크 환경의 고속화가 이루어짐에 따라, 다양한 서비스와 개인의 업무 효율 또한 눈부시게 향상된 반면, 외부 불법 침입자의 공격에 의한 중요 정보 유출 및 조작 등의 피해 역시 급증하게 되었다. 한국 정보보호 진흥원에 따르면, 2001년에는 5,333건, 그리고 2002년에는 전년대비 185%가 증가된 15,192건의 해킹사고가 접수되어 매해 폭발적인 증가세를 보이고 있는 추세이다[1]. 더군다나 인터넷의 대중화로 누구나 해킹 도구를 손쉽게 이용할 수 있게 되면서, 단순 호기심에 의한 공격도 발생하여 그 피해는 더욱 늘어날 전망이다.

이로 인한 인적, 물적 손실을 최소화 하기 위하여 불법 침입에 대비하기 위한 여러 도구와 장비 중 가장 대표적인 것이 침입탐지 시스템이다. 침입탐지 시스템과 관련된 시장은 2001년 2억 9500만불에서 2002년 4억 2천 2000만불로 급격하게 성장하였고 새로운 제품들이 꾸준히 출시되고 있다[4]. 하지만 이와 같은 침입탐지 시스템은 침입 탐지율을 높이는 데에 중점을 두어 개발되고 있기 때문에, 침입의 원인과 경로를 추적하는 기술적인 미비점이 존재한다. 이로 인해, 발생한 침입을 탐지한다고 하더라도 적절한 대응을 할 수 없게 된다. 또한 특정 시스템에 어떠한 침입이 주로 발생하는지에 대한 자료를 얻을 수 없다. 최근, 들어 호스트 기반 침입유형 변형된 공격 방법을 이용하여 침입을 시도하는 횟수가 늘어나고 있으나, 기존 침입탐지 시스템은 이를 잘 탐지하지 못하고 있다. 따라서, 변형된 공격의 침입유형을 판별할 수 있다면, 침입 유형에 알맞은 대응책을 강구 할 수 있을 것이다.

본 논문에서는 솔라리스에서 제공하는 BSM 감사자료의 정상적인 시스템호출 이벤트를 은닉 마르코프 모델(Hidden Markov Model: HMM)로 모델링한다. 그리하여 침입이 탐지 되었을 Viterbi

알고리즘을 이용하여 상태 변화를 분석하고, 경험적으로 얻어진 침입유형별 상태 시퀀스와 의 유사성을 비교하여 침입유형을 판별하는 방법을 제안한다.

2. 침입 유형

호스트 기반에서 침입의 궁극적인 목적은 루트 권한의 획득이며, 가장 대표적인 방법이 버퍼오버플로우(*Buffer Overflow*)이다. 현재 업체 및 기관에서 주로 사용하고 있는 침입유형은 Markus J. Ranum에 의해 분류되었으며, 한국정보보호진흥원 CERTCC-KR에서는 이 분류 방법을 10가지로 확장하였다. 그 중 호스트에서 발생 가능한 유형은 아래 4가지가 있다.

- 버퍼오버플로우: 메모리와 스택의 구조나 OS에 따라 다르기 때문에 다루기가 쉽지 않다. 하지만 현재 인터넷을 통해 버그에 대한 소스코드가 공개되어 있고, 누구나 다운을 받아 운영체제에 맞게 사용할 수 있기 때문에 가장 많이 쓰이고 있는 해킹 방법이다.
- 서비스거부 공격: 시스템이 정상적으로 동작하지 않도록 만드는 공격기법으로, 호스트내의 시스템 자원을 모두 고갈시켜 서비스가 전혀 불가능하게 만든다[2].
- S/W 보안오류: 프로그램 설계 및 구현시의 버그로 인한 취약점을 이용하는 방법으로, 프로그래밍상의 보안 오류와 프로그램 동작상의 보안 오류로 인하여 발생하는 두가지 취약점이 있다.
- 구성설정 오류: 보안을 고려하지 않는 시스템을 구성함으로써 발생하는 오류이며, 시스템 및 시스템에서 제공하는 각종 서비스와 관련된 취약점을 이용한 해킹기법이 포함된다. 이 공격 유형은 해킹을 위해 특별한 해킹 수행 코드가 필요없는 경우가 대부분이다.

3. 제안하는 방법

본 논문에서는 Viterbi 알고리즘을 이용하여 얻어진 각 침입 유형별 상태시퀀스의 평균값과 침입이 발생했을 때의 상태 시퀀스를 유클리드 거리를 이용하여 유사성를 비교한다. 이때, 각 침입 유형과의 거리를 구한 후, 침입 유형과의 값이 가장 작은 것을 발생한 침입 유형이라고 판별한다. 유클리드 거리 공식은 아래와 같으며, d 값이 작을수록 유사성이 크다고 할 수 있다.

$$d = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2}$$

제안하는 시스템의 구조는 그림 1과 같다.

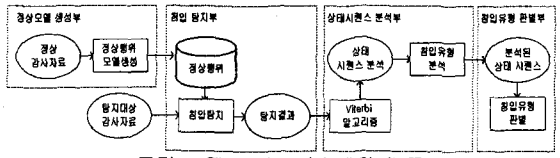


그림1. 제안하는 시스템의 구조

3.1 HMM 기반의 침입 탐지 시스템

솔라리스에서 제공하는 기본 보안 모듈(Basic Security Module: BSM)을 이용한 감사데이터는 운영체제에서 발생한 시스템 호출 이벤트들과 그에 관련된 사용자 및 프로세스 정보를 포함하는 자료로 침입탐지 시스템에서 가장 많이 쓰이는 척도이다. 본 논문에서는 기존 연구된 음성인식이나 영상인식 분야에서 널리 쓰이는 HMM을 기반으로 한 침입 탐지 시스템을 사용하였는데 이는 HMM이 실제적인 생성 경위를 알기 힘든 이벤트 시퀀스를 잘 모델링 할 수 있는 방법으로 시스템 호출 감사자료를 이용하는 데에 유용하기 때문이다[3].

HMM은 상태 전이 확률분포 A, 관측기호 확률분포 B, 그리고 초기상태 확률 분포 π로 구성되며, 일반적 기호로는 $\lambda = (A, B, \pi)$로 나타낸다[4]. 정상모델 생성부에서는 구축된 모델 λ로부터 주어진 시퀀스가 나올 수 있는 확률 값인 $P(O|\lambda)$가 최대가 되도록 HMM의 구성요소들을 조정해 나가며 정상행위 감사자료를 모델링한다.

이렇게 생성된 모델을 이용하여, 침입 탐지부에서는 침입이 들어있는 시스템호출 감사자료를 이용하여 입력하고, 정상행위 모델에서 입력된 행위가 나올 확률을 계산한 후, 이 확률값이 정상행위 모델링에서 구한 임계값보다 낮으면 침입으로 판정한다.

3.2 Viterbi 알고리즘을 이용한 공격유형 판별시스템

앞선 침입탐지 모듈에서 침입이라고 판정되면, 그 시점의 시스템호출 이벤트의 상태 시퀀스를 알아내기 위해 상태 시퀀스 분석부에 Viterbi 알고리즘을 이용한다. Viterbi 알고리즘은 주어진 상태 시퀀스에서 가장 높은 확률을 가진 상태전이 경로를 찾아주는 것으로서[5], HMM으로 모델링된 음성이나 문자를 인식하기 위한 시스템에도 많이 이용되고 있다[6]. 본 논문에서는 HMM 기반 정상 행위 모델에 시스템 호출 시퀀스를 입력으로 넣어, 각 정상행위에서 현재

행위가 생성되었을 확률이 가장 높은 상태 시퀀스를 구하기 위해 사용한다. Viterbi 알고리즘은 t시간에서 상태 i에 있을 확률인 $\delta_t(i)$, 시간 t일 때 상태 j로 전이될 가장 높은 확률의 상태를 가르키는 $\Psi_t(j)$, 그리고 가장 높은 확률의 상태시퀀스를 가진 s_t^*의 변수를 구성한다.

여기서, 주어진 상태 시퀀스 $O=O_1,O_2,...,O_T$를 시스템 호출 이벤트로 매칭한다. 그리고 시간 t를 변경해가면서 학습된 모델에서 구한 상태 전이 확률과 함께 계산하여, 그 중 가장 높은 확률을 가지는 값으로 분석 후 나온 상태 시퀀스로, 경험적으로 얻어진 공격유형별 상태 시퀀스와 비교하여 그 중 가장 최저의 유클리드 거리값을 가지는 상태 시퀀스가 해당되는 공격유형으로 판별된다.

본 실험에 이용한 Viterbi 알고리즘은 다음의 과정을 거치게 된다.

□ 1단계(초기화):

$$\delta_1(i) = \pi_i b_i(O_1), \quad 1 \le i \le N$$

$$\Psi_q(i) = 0$$

□ 2단계(순환):

$$\delta_t(j) = \max_i \left[\delta_{t-1}(i)a_{ij}\right] b_j(O_t), \quad 2 \le t \le T, 1 \le j \le N$$

$$\Psi_t(j) = \arg\max_i \left[\delta_{t-1}(i)a_{ij}\right] b_j(O_t), \quad 2 \le t \le T, 1 \le j \le N$$

□ 3단계(종료):

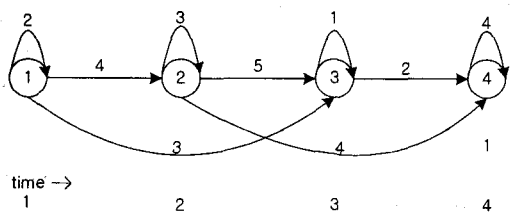
$$P^* = \max_{s \in S_F} \left[\delta_T(s)\right]$$

$$S_T^* = \arg\max_{s \in S_F} \left[\delta_T(s)\right]$$

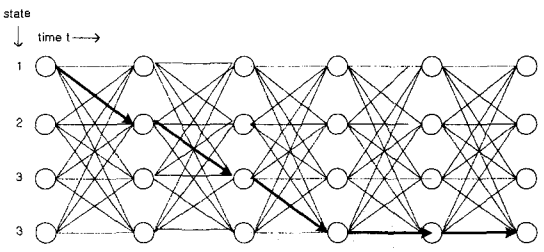
□ 4단계(역추적):

$$s_t^* = \Psi_{t+1}(s_{t+1}^*), \quad t = T-1, T-2, \cdots, 1$$

그림 2는 상태수가 3인 HMM에서의 Viterbi 알고리즘이 동작되는 원리를 간단하게 보여주고 있다.



(a) 상태 기계



(b) Viterbi 알고리즘의 예

그림 2. 상태수가 4인 HMM에서의 Viterbi 알고리즘 작동원리

그림 2.(a)와 같이 초기상태 1에서 다음 상태로 전이될 확률이 가장 높은 것을 선택하면 상태 2가 된다. 그리고 상태 2에서 다음 상태로 전이될 가장 높은 확률을 가진 상태는 3이 되며 이런 과정을 시간 T만큼 반복하면 상태시퀀스가 그림 2(b)와 같은 {1,2,3,4,4,4}의 상태 시퀀스를 가지게 된다.

4. 실험 결과

우선 정상행위를 모델링을 하는데에 6명의 사용자가 보름동안 16,470개의 명령어를 입력하여 발생한 160,448개의 이벤트가 수집된 13MB의 감사자료를 사용하였으며, 탐지 대상 감사 자료에 쓰인 공격유형은 버퍼오버플로우와 서비스 거부 공격을 사용하였다.

버퍼오버플로우 공격은 시스템의 취약점을 이용하여 루트 권한을 획득하는 방법으로, 가장 많이 쓰이는 침입 유형이며, 서비스 거부 공격은, 특정 시스템을 대상으로 다른 프로세스들에게 올바른 서비스를 제공하지 못하게 하는데, 이 두 가지가 호스트 기반에서 발생할 수 있는 가장 대표적인 공격 유형이므로 본 논문에서는 이 두 가지를 대상으로 침입유형을 판별하는 실험을 하였다. 각 침입 유형과 침입 횟수는 표 1과 같다.

표 1. 침입 유형 및 침입 횟수

유형	침입형태	횟수
Buffer Overflow	Open View xlock Heap Overflow	8
	Lpset –r Buffer Overflow Vulnerability	7
	Kcms_configure KCMS_PROFILES	4
Deinal Of Service	디스크 채우기	9
	메모리 고갈	9
	프로세스 테이블 고갈	7

공격 유형별 상태 변화를 비교하는 대상으로, 공격 유형별로 각각 30번의 실험으로 얻어진 평균값으로 이용하였으며, 그림 3은 각 공격별 상태 시퀀스의 평균값을 나타낸다.

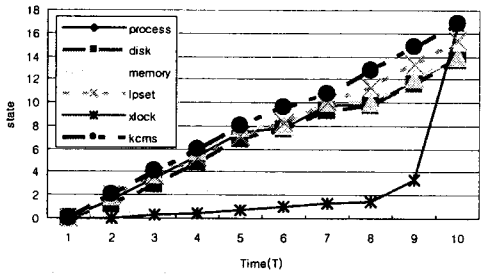


그림 3. 공격 유형별 상태 변화

HMM의 상태수와 한번에 입력될 시스템호출 이밴트 길이를 여러가지로 변화시켜 HMM 구성을 바꾸며 실험을 반복하였다. 그 중에서 100%의 침입 탐지율과 최소의 탐지 오류율을 내는 HMM 임계값 -20.83에서 침입 유형 판별 실험을 하였으며, 실험 결과는 표2와 같다.

표2. 상태수 20, 관찰길이 10 에서의 실험 결과

	A	B	C	D	E	F	Rate
A	7	1	-	-	-	-	88%
B	-	3	2	1	1	-	42%
C	-	1	3	-	-	-	75%
D	-	-	-	3		6	33%
E	-	-	-	4		5	0%
F	-	-	-	-	1	6	86%

A는 xlock 침입, B는 lpset 침입, C는 kcms_sparc 침입, D는 process 테이블 채우기 공격, E는 디스크 채우기 공격 그리고 F는 메모리 고갈 공격을 나타낸다. 실험 결과, xlock과 메모리 고갈 공격은 상대적으로 판별률이 뛰어났다. 하지만 lpset 공격, process 테이블 채우기 공격 그리고 디스크 채우기 공격의 판별율은 현저하게 떨어짐을 알 수 있었다.

이 중, 프로세스 테이블 채우기 공격과 디스크 채우기 공격의 판별률이 떨어지는 이유는 이 둘의 상태시퀀스가 메모리 고갈 공격의 상태시퀀스와 매우 흡사하기 때문이다. 실제 이 둘의 침입을 50% 이상 메모리 고갈 공격이라고 판별하였다.

또한, HMM의 주요 변수 중 상태수를 5, 10, 15, 20으로 변화해가며 실험을 수행하였으며, 이 중 상태수가 20 이상일 경우에만 침입유형을 판별할 수 있었다. 실제 상태수 20 미만에서의 상태시퀀스를 분석한 결과, 실험에 사용된 6가지 공격 유형 모두의 상태시퀀스가 동일하게 나왔다. 이러한 원인은 상태수가 많을수록 복잡도가 높아지며, 다른 상태로 전이될 수 있는 경우의 수가 그만큼 많아지기 때문이다. 반면, 상태수가 적은 경우는 다른 상태로 전이될 경우의 수가 많지 않으므로, 서로 다른 침입이라고 하더라도, 상태시퀀스가 동일해질 확률이 높아진다.

5. 결론

본 논문에서 HMM기반의 비정상행위 침입탐지 시스템에서 침입유형을 판별하기 위한 방법을 제안하고, 그 가능성을 밝히기 위해 상태수를 5, 10, 15, 20으로 각각 변화해가며 실험을 하였다. 실험 결과, xlock 공격, kcms_sparc 공격, 메모리 고갈 공격에 대해서는 비교적 정확하게 판별할 수 있었지만, 나머지 lpset, process 테이블 채우기 그리고 디스크 채우기 등의 공격에 대해서는 침입 유형을 올바르게 판별하지 못하였다. 이 중, Process 테이블 채우기 공격과 디스크 채우기 공격의 경우 메모리 고갈 공격과 매우 흡사한 상태 시퀀스를 가지기 때문에 낮은 판별률을 보였다.

또한, 침입유형을 판별하기 위해서는 최소 20개 이상의 상태수가 필요함을 알 수 있었다. 하지만, 상태수가 많아짐에 따라 결과를 알아내는 데에 걸리는 시간이 많이 소요되므로, 적은 상태수에서 침입 유형을 알아낼 수 있는 방법에 대한 연구와 process 테이블 공격, 디스크 채우기 공격 및 lpset 공격을 좀더 정확하게 판별할 수 있는 방법에 대한 연구가 필요하다.

감사의 글

본 연구는 대학 IT 연구센터 육성/지원 사업의 연구 결과로 수행되었음.

참고문헌

[1] CERTCC-KR, 한국 정보 보호진흥원, http://www.certcc.or.kr, 2003.

[2] F. Lau, S. H. Rubin, M. H. Smith and L. Trajkovic, " Distributed denial of service attacks," *In Proc. of IEEE Conference on Systems, Man and Cybernetics*, pp. 2275-2280, 2000.

[3] S.-B. Cho and H.-J. Park, "Efficient anomaly detection by modeling privilege flows using hidden Markov model," *Computers & Security*, vol. 22, no. 1, pp. 45-55, 2003.

[4] L.R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proc. of IEEE*, vol. 77, no. 2, pp.257-286, February 1989.

[5] G. D. Forney Jr. " Maximum-likelihood sequence detection in the presence of intersymbol interference," *IEEE Transactions on Information Theory*, vol. 18, no. 30, pp.363-378, May 1972.

[6] L. R. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition*, Prentice Hall, Englewood Cliffs, New Jersey, 1993. Chapter 6.