은닉 마르코프 모델을 이용한 침입탐지 시스템의 모델링 성능분석

최종호, 조성배
연세대학교 컴퓨터과학과
e-mail:[hosoft,sbcho]@candy.yonsei.ac.kr

Analysis of Modeling Power of Intrusion Detection System Using Hidden Markov Model

Jongho Choy and Sung-Bae Cho
Computer Science Department, Yonsei University

요약

정보통신 구조의 확산과 함께 전산시스템에 대한 침입과 피해가 증가되고 있으며 침입탐지 시스템에 대한 관심과 연구가 늘어나고 있다. 본 논문에서는 HMM(Hidden Markov Model)을 이용하여 사용자의 정상행위에서 생성된 시스템 호출정보를 모델링한 후 사용자의 비정상행위를 탐지하는 침입탐지 기법을 제안한다. 실험을 통해 침입탐지를 위한 최적의 HMM 매개변수를 결정하고, 사용자 구분이 없는 단일모델링, 사용자별 모델링, 사용자 그룹별 모델링 방식의 정상행위 모델링을 평가하였다. 실험결과 신뢰도 높은 침입탐지 시스템의 구축을 위해서는 보다 정교한 모델의 클러스터링이 필요함을 알 수 있었다.

1. 서 론

침입탐지는 불법적인 사용이나 오용, 남용 등에 의한 침입을 알아내는 것으로[3][6], 침입에 의한 피해의 규모와 횟수가 증가함에 따라 효과적인 침입탐지 시스템에 대한 요구가 높아지고 있다.

침입탐지 방법은 크게 오용탐지와 비정상행위탐지로 나뉜다. 오용탐지 기법은 알려진 공격에 대한 정보를 구축한 후 사용자나 시스템 또는 프로그램의 현재 행동이 공격패턴과 일치하는지를 검사한다. 공격패턴 정보를 가지고 있으므로 정상행위를 공격행위로 간주하는 오류(false-positive error)가 낮고, 공격패턴만 검색하면 되므로 경제적인 장점이 있는 반면 알려지지 않은 새로운 공격은 탐지할 수 없다는 단점을 가지고 있다. 이에 비해 비정상행위탐지 기법은 모델링된 정상행위에서 벗어나는 행동은 공격행위로 간주하기 때문에 공격행위를 정상행위로 간주하는 오류(false-negative)가 낮다. 그러나 정상행위 모델링을 위해서 다량의 데이터를 분석해야 하므로 구현비용이 높고, 학습되지 않은 정상행위는 비정상행위로 간주되므로 정상행위가 공격행위로 간주되는 오류(false-positive)가 높다.

본 논문에서는 비정상행위탐지 방식을 사용한 침입탐지 시스템을 구현한다. Lane과 Broadly[5]가 지적했듯이 사람과 컴퓨터간의 상호작용은 인과관계라는 관찰에 주목하여 사용자가 생성하는 이벤트의 시퀀스 정보를 사용하는 비정상행위탐지 기법을 구현한다. 이벤트 시퀀스 정보로는 Sun사의 BSM(Basic Security Module)을 통해 획득한 감사자료 중 시스템 호출을 사용하며 모델링 및 판정기법에는 음성인식 및 여러 분야에서 소스가 알려지지 않은 대상을 모델링하는데 널리 사용되고 있는 HMM을 사용하였다.

2. 관련연구

오용탐지를 위해서는 참조되는 정상모델과 모델링 기법 및 테스트 데이터가 정상에서 벗어났는지 여부를 알아내기 위한 추론기법이 있어야 한다.

정상모델구축을 위해 모델링하는 대상은 크게 두가지로서 하나는 사용자이며 또 하나는 프로그램이다. 사용자 모델링[4]은 해당 사용자의 정상적인 사용패턴을 모델링하며

프로그램 모델링[2][8]은 프로그램이 정상적으로 사용되기 위해서 발생되어야 하는 이벤트 및 정보의 변화를 모델링 한다.

모델링 및 추론을 위해서는 다양한 기법[1]들이 사용된다. 가장 널리 사용되는 접근방법은 통계적 기법이다. 통계적 기법은 사용자나 시스템 행동을 시간에 따라 샘플링된 여러 가지 변수들에 의해 측정하여 평균과 표준편차로 모델링한다. 탐지시에 변수의 표준편차에 기반해서 관찰데이터가 임계값을 넘어섰는지를 검사한다. 전문가 시스템은 정상행위를 규칙의 집합으로 표현한다. 탐지시에는 현재 활동을 구축된 규칙들과 비교하는 방식으로 비정상행위 여부를 가린다. 이외에 신경망의 학습능력과 일반성을 비정상행위 탐지에 이용하기도 하며, 사용자의 의도 파악을 통해서 고수준에서 사용자의 행위를 모델링하고 예측하기도 하며 컴퓨터 면역학을 응용하기도 한다.

컴퓨터 내의 프로그램 수행이나 사용자 활동은 시간에 따라 순서적으로 발생한다는 점은 자연스럽게 모델링에서 순서적인 정보를 이용하려는 시도들로 이어졌다. [2]는 부분적으로 동등정합(equality matching) 기법을 사용한다. 시간적으로 상호 위치한 이벤트상에 시그너쳐를 남기는 경향의 공격특성을 이용하기 위하여 이벤트들을 고정크기 윈도우로 구획한 후 구축되어있는 정상 시퀀스들과 비교해서 비정상행위 계수기를 통해 추적한다. 계수기는 각 고정윈도우에서 시스템호출을 계수하는 계수기에서부터 비정상윈도우를 계수하는 계수기까지 다양한 수준에서 사용되며, 각 수준에서는 어느 선에서 상위수준으로 비정상행위를 전파할지를 결정하는 임계값들이 적용된다. 충분한 수의 시스템 호출 윈도우가 비정상적이면 침입탐지 플래그를 설정한다.

Elman 신경망을 이용한 비정상행위 탐지[2]는 시퀀스간 상태정보를 유지할 수 있는 신경망 기법으로 신경망의 학습과 일반화 능력을 유지하면서 순서적 특성을 반영하려고 시도한 것이다. Elman 신경망은 일반적인 입력, 출력, 은닉 노드에 추가적으로 문맥 노드를 가지고 있으면서 입력간의 상태정보를 유지하는데 사용한다.

Warrender 등[7]은 유한상태기계의 맥락에서 HMM을 사용하여 순서정보를 표현하려 시도하였다. 일반적인 HMM 적용방식인 시퀀스를 직접 평가하지 않고 학습된 HMM 그래프를 비결정형 유한오토마타로 해석해서 각 이벤트 시퀀스의 시스템 호출마다 가장 가능한 전이와 심볼을 추측하여 관찰 이벤트 시퀀스의 전이와 심볼과 비교한다.

3. HMM을 사용한 침입탐지

3.1. 침입탐지 시스템 개요

본 논문에서 개발하는 침입탐지시스템은 그림 0과 같이 데이터 필터링과 데이터 축약을 담당하는 전처리 모듈과 정상행위 모델링과 추론 및 판정을 담당하는 비정상행위 판정모듈로 구성된다. 모델학습을 통해 정상행위를 프로파일 데이터베이스로 구축하여 판정시 사용한다.

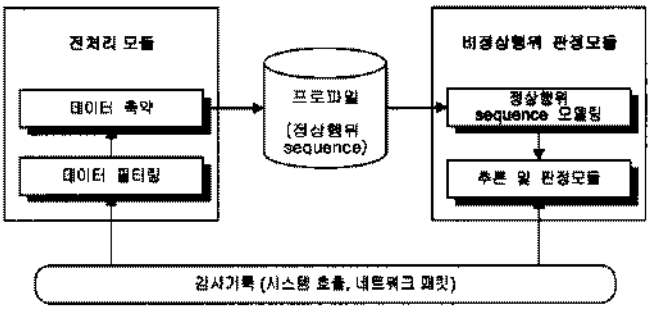

그림 1. 침입탐지 시스템 개요

3.2. 전처리

사용자 감사기록으로는 사용자의 시스템 호출번호를 사용하였다. 모든 시스템호출번호가 다 사용되지 않으므로 통계적으로 빈도가 높은 49개의 시스템호출에 대해 0부터 48번까지의 번호를 부여하였고 그 밖의 시스템 호출은 49번을 부여하여 총 50개의 축약된 시스템 호출번호를 사용하였다. 순서적으로 생성되는 시스템 호출번호는 일정 크기의 윈도우를 옆으로 이동시켜가면서 윈도우 크기 만한 시퀀스로 추출하였다.

번호	호출	번호	호출	번호	호출
0	exit	17	mkdir	34	munmap
1	fork	18	rmdir	35	setegid
2	create	19	setrlimit	36	seteuid
3	link	20	pathconf	37	putmsg
4	unlink	21	open_r	38	getmsg
5	chdir	22	open_w	39	audition_setcond
6	chmod	23	open_wc	40	statvfs
7	chown	24	open_rw	41	sysinfo
8	kill	25	open_rwc	42	fork1
9	symlink	26	close	43	sockconnect
10	readlink	27	getaudit	44	login
11	execve	28	setaudit	45	logout
12	vfork	29	ioctl	46	telnet
13	getgroups	30	setuid	47	rlogin
14	setpgrp	31	utime	48	su
15	fcntl	32	nice	49	etc.
16	rename	33	setgid		

표 1. 축약된 시스템호출

3.3. 침입탐지를 위한 HMM

HMM은 실제적인 생성모델을 알 수 없고 단지 생성된 시퀀스에 의해서만 확률적으로 관측할 수 있는 이중으로 확률적인 절차로서[7], 사용자의 행위시퀀스를 모델링하기에 유용한 도구이다. 이 모델은 관찰 시퀀스의 길이, 상태

수, 심볼수와 학습에 의해 조정되는 전이확률, 관측확률, 초기상태분포로 구성이 된다. 전이확률은 한 상태에서 다음상태로 전이할 확률을 나타내며, 관측확률은 한 상태에서 특정 심볼이 관측될 확률을 나타낸다. 초기 상태 분포는 처음에 해당 상태에서 시작할 확률을 나타낸다. HMM은 다음과 같이 표현되며, 모델 λ는 간략히 (A, B, π)로 표현될 수 있다.

- T : 관찰 시퀀스의 길이
- N : 모델의 상태수
- M : 관찰 심볼의 수
- $Q = q_1, q_2, \ldots, q_N$: 상태들
- $V = v_1, v_2, \ldots, v_M$: 가능한 관찰심볼의 이산적인 집합
- $A = \{a_{ij}\}, a_{ij} = \Pr(q_j \text{ at } t+1 \mid q_i \text{ at } t)$: 상태전이 확률분포
- $B = \{b_j(k)\}, b_j(k) = \Pr(v_k \text{ at } t \mid q_j \text{ at } t)$: 관측 심볼 확률분포
- $\pi = \{\pi_i\}, \pi_i = \Pr(q_i \text{ at } t=1)$: 초기 상태 분포

정상행위 모델링과 비정상행위 판정은 그림 2와 같은 과정을 통해서 수행된다.

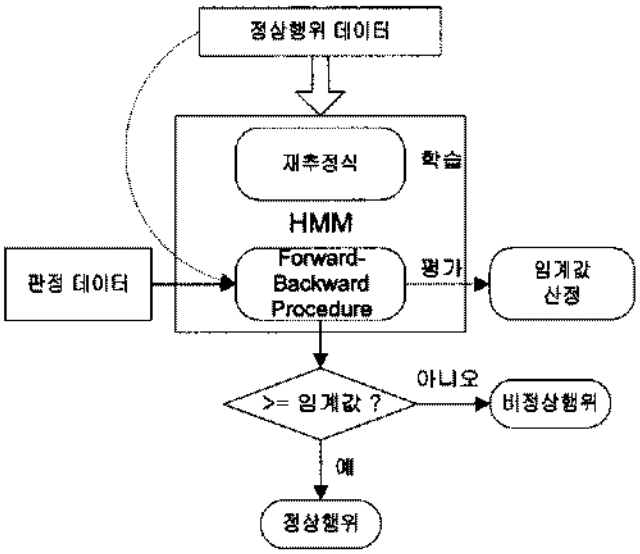

그림 2. HMM을 이용한 비정상행위 판정과 정상행위 모델링과정

가) 비정상행위 판정

비정상행위 판정에서는 이미 구축되어 있는 정상행위별 HMM에 사용자행위시퀀스를 입력으로 넣어 각 정상행위에서 현재 행위가 생성되었을 확률을 구한다. 확률을 구하는 방법으로는 forward- backward procedure나 Viterbi 알고리즘을 사용할 수 있다[7]. 각 모델별로 구해진 확률은 판정모듈에 전달되어 비정상행위인지 판정한다.

forward-backward procedure에서는 forward 변수인 α와 backward 변수인 β를 사용해서 입력시퀀스 O가 해당 모델 λ로부터 나왔을 확률 $\Pr(O|\lambda)$를 계산한다. forward 변수 α는 시간 t에 부분관찰 시퀀스 $O_1, O_2, \ldots, O_t$를 보고 상태 q_i에 있을 확률로 다음과 같이 정의된다.

$$\alpha_t(i) = \Pr(O_1, O_2, \ldots, O_t, i_t = q_i \mid \lambda)$$

이 정의에 따르면 $\alpha_T(i)$는 입력시퀀스 O의 모든 심볼을 순서에 맞게 가지고 있으면서 최종상태가 i인 확률을 나타낸다. $\alpha_T(i)$를 모든 상태 i에 대해 고려하면 $\Pr(O|\lambda) = \Pr(O_1, O_2, \ldots, O_T \mid \lambda)$를 구할 수 있다. $\alpha_t(i)$는 다음 절차에 의해 귀납적으로 구할 수 있다.

- 단계 1 (초기화) :

$$\alpha_1(i) = \pi_i b_i(O_1)$$

- 단계 2 (귀납) :

$$\text{for} \quad t=1 \text{ to } T-1$$
$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i)\, a_{ij} \right] b_j(O_{t+1})$$

- 단계 3 (종료) :

$$\Pr(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

backward 변수 $\beta_t(i)$는 다음과 같이 정의되며 forward변수를 구하는 것과 유사한 과정에 의해서 구할 수 있다.

$$\beta_t(i) = \Pr(O_{t+1}, O_{t+2}, \ldots, O_T \mid i_t = q_i, \lambda)$$

- 단계 1 (초기화) :

$$\beta_T(i) = 1$$

- 단계 2 (귀납) :

$$\text{for} \quad t = T-1 \text{ to } 1$$
$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)$$

나) 정상행위 모델링

정상행위 모델링은 전처리 단계에서 생성된 정상행위 시퀀스를 기반으로 HMM의 매개변수를 결정하는 과정이다. HMM의 매개변수 결정은 주어진 시퀀스 O가 해당 모델 λ로부터 나왔을 확률인 $\Pr(O|\lambda)$값이 최대가 되도록 $\lambda = (A, B, \pi)$를 조정한다. 이를 계산하는 해석적인 방법은 알려져있지 않고 반복적으로 λ를 결정하는 방법으로 Baum-Welch의 재추정식이 있다[7].

Baum-Welch 재추정식에서는 두 개의 변수가 추가로 사용된다. $\xi_t(i,j)$는 시간 t에 상태 q_i에 있다가 시간 $t+1$에 상태 q_j에 있을 확률로 정의되며 다음과 같이 표현될 수 있다.

$$\xi_t(i,j) = \Pr(i_t = q_i, i_{t+1} = q_j \mid O,\lambda) = \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{\Pr(O|\lambda)}$$

$\gamma_t(i)$는 시간 t에 상태 q_i에 있을 확률이며 다음 수식을 통해 구할 수 있다.

$$\gamma_t(i) = \sum_{j=1}^{N}\xi_t(i,j)$$

두 값을 시간 t에 대해 각각 합하면 하나의 시퀀스에서 각각 상태 i에서 j로 변할 기대값과 상태 i에 있을 기대값을 구할 수 있다. 위의 값이 구해지면 다음 수식에 의해서 새 모델 $\overline{\lambda} = (\overline{a}, \overline{b}, \overline{\pi})$를 구할 수 있다.

$$\overline{\pi_i} = \text{시간 } (t=1)\text{에 상태 } S_i \text{ 에 있을 확률} = \gamma_1(i)$$

$$\overline{a_{ij}} = \frac{\text{상태 } S_i\text{에서 상태 } S_j\text{로 전이할 기대횟수}}{\text{상태 } S_i\text{에서 전이할 기대횟수}} = \frac{\sum_{t=1}^{T-1}\xi_t(i,j)}{\sum_{t=1}^{T-1}\gamma_t(i)}$$

$$\overline{b_j(k)} = \frac{\text{상태 } j\text{에서 심볼 } v_k \text{ 를 볼 기대횟수}}{\text{상태 } j \text{ 에 있을 기대횟수}} = \frac{\sum_{t=1 s.t. O_t=v_k}^{T}\gamma_t(j)}{\sum_{t=1}^{T}\gamma_t(j)}$$

시퀀스 O를 관찰한 결과로 $\overline{\lambda}$를 구한 후 $\Pr(O|\lambda)$와 $\Pr(O|\overline{\lambda})$를 비교한다. $\Pr(O|\lambda)$가 더 크다면 우도 함수의 임계점에 다다랐으므로 재추정 과정을 종료한다. $\Pr(O|\overline{\lambda})$가 더 큰 경우는 더 나은 모델이 생성된 경우이기 때문에 λ를 $\overline{\lambda}$로 대치한 후 재추정 과정을 반복한다.

4. 실험결과

실험 데이터로는 3명의 사용자가 1주일간 발생시킨 데이터를 사용하였다. 주사용 프로그램은 문서편집기와 컴파일러, 그리고 사용자가 작성한 프로그램이었다. 학습 데이터와 테스트 데이터는 사용자별로 각각 10,000개, 즉 60,000개를 사용했으며, 테스트 데이터에는 각 사용자별로 u2r 침입을 17차례 넣었다. HMM의 모델은 순서정보를 잘 표현한다는 left-to-right 모델을 사용했다.

먼저, 첫 번째 실험은 전체 사용자 데이터를 단일 모델을 사용해서 수행하였다. 그림 3은 침입행위가 발생한 경우의 시퀀스평가값의 변화를 보여준다. 침입행위가 시작된 시간 11에서부터 침입행위가 종료된 시간 32사이에 있는 시퀀스들의 평가값이 급격히 낮아져 HMM이 효과적으로 침입을 탐지하고 있는 것을 보여주고 있다.

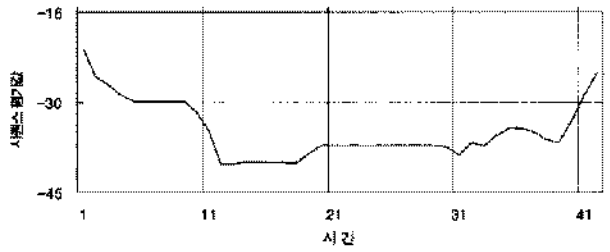

그림 3. 침입행위 발생시 평가값의 변화

이 실험에서는 침입탐지에 최적의 성능을 보일 수 있는 HMM의 매개변수를 결정하기 위한 실험도 병행하였다. 그림 4는 상태수의 변화에 따른 ROC(Receiver Operating Characteristic) 곡선으로서[2] 변경 가능한 매개변수의 조정에 따른 탐지율과 false-positive 오류율의 변화를 보여주고 있다. 본 실험에서는 임계값을 조정해서 ROC 곡선을 얻었다. 바람직한 침입탐지시스템은 낮은 false-positive 오류에서 높은 침입탐지율을 보여주어야 하므로 곡선이 왼쪽에 있을수록 좋은 성능을 나타낸다. 상태수를 5, 10, 15, 30으로 변경해가면서 ROC곡선을 얻은 결과 그림 4와 5에서 볼 수 있듯이 상태수 10에서 가장 낮은 false-positive 오류를 보여 가장 좋은 결과를 얻을 수 있었다.

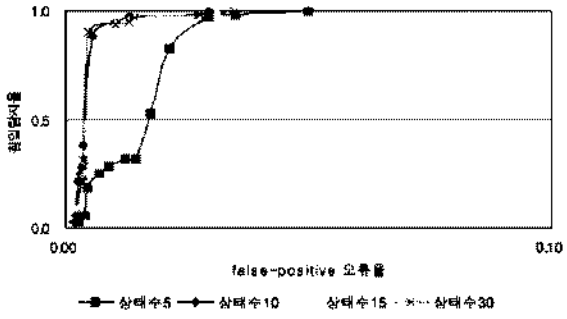

그림 4. 상태수에 따른 ROC곡선(단일모델)

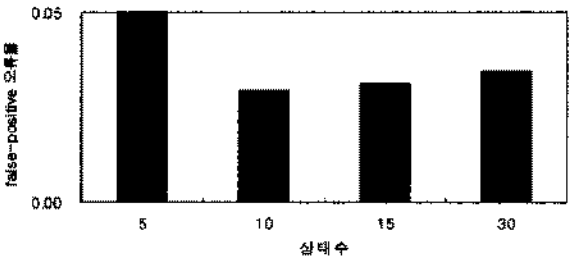

그림 5. 상태수에 따른 false-postive오류율(단일모델)

그림 8과 9는 시퀀스 길이를 변경해가면서 시스템의 성능을 평가한 경우를 보여준다. 시퀀스 길이 30에서 가장 좋은 성능을 내는 것을 볼 수 있다.

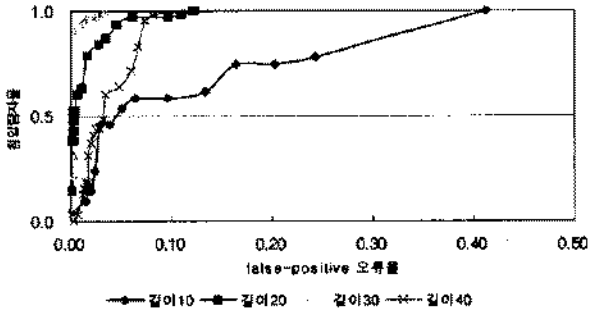

그림 6. 시퀀스길이에 따른 ROC곡선(단일모델)

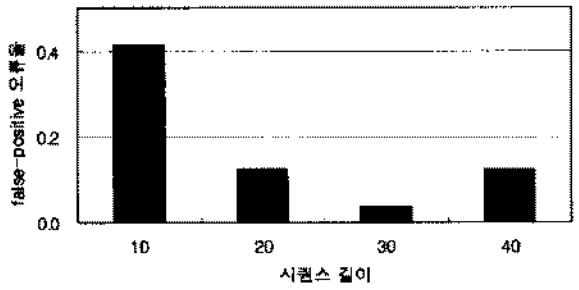

그림 7. 시퀀스길이에 따른 false-positive오류율(단일모델)

첫번째 실험에서 침입탐지에 HMM 적용의 타당성을 검토하고 HMM의 최적의 매개변수를 결정한 후, 각 사용자별로 별도의 모델을 만들어서 두번째 실험을 수행하였다.

표 2는 각 모델별 탐지율과 false-positive 오류율을 보여준다. 모든 테스트 데이터를 단일 모델로 모델링한 경우보다 각 사용자별로 모델링한 경우에 전반적으로 더 낮은 false-positive 오류율을 보여준다.

	단일 모델링	사용자별 모델링		
		사용자 1	사용자 2	사용자 3
정상행위 평균(편차)	-14.74 (10.42)	-15.54 (5.88)	-4.72 (8.42)	-14.25 (12.34)
임 계 값	-32.45	-26.21	-31.35	-36.92
탐 지 율	100%	100%	100%	100%
false-positive 오류율	2.95%	4.31%	1.73%	1.96%

표 2. 사용자별 모델링에 따른 비정상행위 탐지결과

그림 8의 ROC 곡선에서도 단일 모델링의 경우보다 세분화된 모델링의 경우가 전반적으로 좋은 성능을 보여줌을 확인할 수 있다.

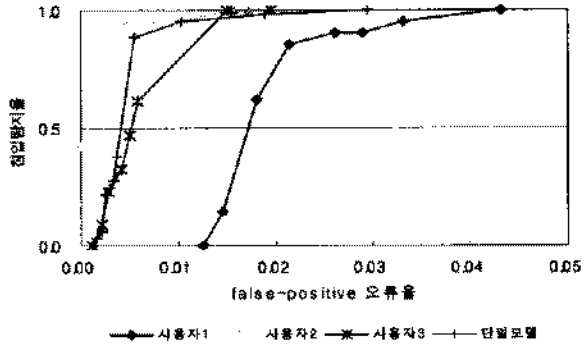

그림 8. ROC 곡선(모델비교)

정상행위를 모델링한 사용자와 판정사용자를 다르게 한 경우 표 3과 같이 시퀀스평가값이 낮게 나왔다.그림 9, 10, 11에서는 사용자가 달라진 경우 false-positive 오류율이 증가함을 확인할 수 있다. 이는 본 기법이 사용자의 정상행위를 모델링하는데 효과적임을 보여주며 다른 사용자의 불법적인 계정획득에 의한 침입행위 탐지에 효과적으로 사용될 수 있음을 보여준다.

		평가 사용자		
		사용자1	사용자2	사용자3
모델링 사용자	사용자1	0	366	55
	사용자2	566	315	336
	사용자3	475	346	0

표 3. 시퀀스평가값이 -50이하인 시퀀스수

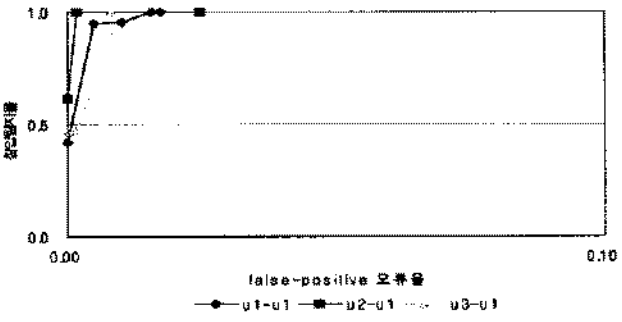

그림 9. 정상모델변화에 따른 ROC곡선(사용자1)

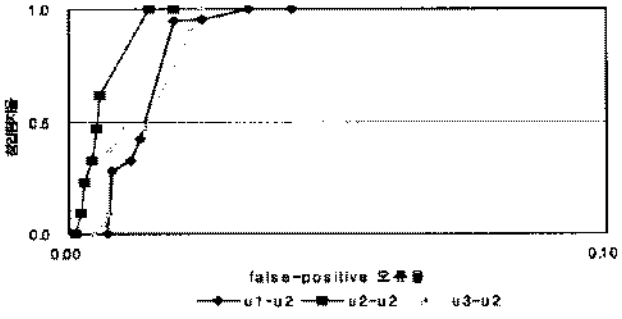

그림 10. 정상모델변화에 따른 ROC곡선(사용자2)

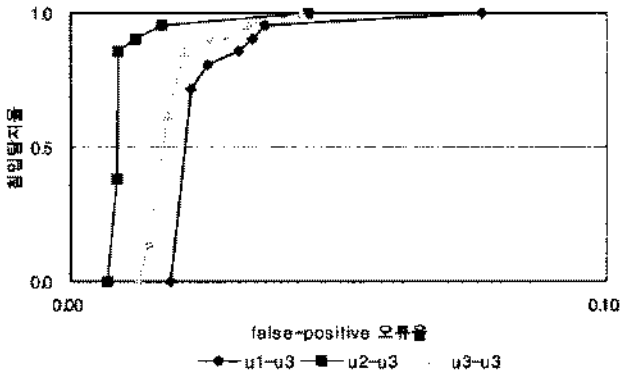

그림 11. 정상모델변화에 따른 ROC곡선(사용자3)

마지막 실험은 비슷한 행동을 보이는 사용자들을 그룹으로 묶어서 그룹별로 모델링했다. 이는 모델링할 사용자의 증가에 따른 정상행위 모델링 비용을 감소시키려는 것과 유사한 행동을 보일 것으로 기대되는 사용자들의 행위데이터를 통합함으로써 해당 사용자 그룹의 다양한 정상행위를 수집하려는 것이다. 그룹 선정시에는 사용자1과 3이 동일 과제를 수행하고 있다는 것과 표 3에서 보듯이 두 사용자 사이의 행위차가 상대적으로 작다는 점을 고려하였다.

표 4의 결과를 보면 사용자 1, 3을 그룹으로 묶은 경우 사용자 1의 오류율은 개선되었지만 사용자 3의 오류율은 나빠졌다. 실험대상 사용자집단의 크기가 유사성을 찾을 만큼 충분히 크지 않아서 상대적으로 유사한 두 사용자의 행위 유사성이 기대만큼 크지 않았던 것으로 생각된다.

	사용자별 모델링			그룹 모델링 (사용자1,3)
	사용자 1	사용자 2	사용자 3	
정상행위 평균(편차)	-15.54 (5.88)	-4.72 (8.42)	-14.25 (12.34)	-16.20 (8.70)
임 계 값	-26.21	-31.35	-36.92	-27.83
탐 지 율	100%	100%	100%	100%
false-positive 오류율	4.31%	1.73%	1.96%	4.09%

표 4. 사용자별 모델링에 따른 비정상행위 탐지결과

5. 결론 및 향후연구

본 논문에서는 HMM을 사용하여 사용자의 정상행위를 모델링한 후 순서적으로 생성되는 이벤트를 분석하여 비정상행위를 판정하는 컴퓨터 침입탐지기법을 제안하였다. 본 기법은 침입발생시 적절하게 이를 탐지하였으며, 모델링된 사용자외의 다른 사용자에 의한 사용도 탐지하였다. false-positive 오류가 발생한 부분은 학습시 정상행위로 모델링되지 않은 것이었다. 포괄적으로 정상행위 데이터를 제공할 수 있다면 HMM이 효과적인 침입탐지기법으로 사용될 수 있음을 알 수 있었다.

모델간 비교를 통해서는 하나의 모델로 모델링한 것보다 사용자별로 모델링한 경우가 오류가 더 적음을 확인하였다. 사용자 그룹별로 모델링한 경우에는 성능의 개선을 보여주지 못하였지만 사용자집단의 크기가 충분히 크고 사용자별, 행위별 유사성 판정 방식을 개선한다면 처리성능 및 모델링 능력을 향상시킬 수 있을 것으로 보인다.

앞으로 변별력있는 사용자 이벤트 추출에 관한 연구와 사용자 및 사용패턴간의 유사성 판정에 관한 연구가 수행되어야 할 것이다.

참고문헌

[1] H. Deba, M. Dacier, A. Wespi, "Towards a taxonomy of intrusion-detection systems," *Computer Networks*, vol. 31, pp. 805-822, 1999.

[2] A. K. Ghosh, A. Schwartzbard and M. Schatz, "Learning program behavior profiles for intrusion detection," *Proc. Workshop on Intrusion Detection and Network Monitoring*, Santa Clara, USA, April 1999.

[3] S. Kumar and E. H. Spafford, "An application of pattern matching in intrusion detection," *Technical Report CSD-TR-94-013*, 1994.

[4] T. Lane and C. E. Broadly, "Temporal sequence learning and data reduction for anomaly detection," *Proc. ACCS '98*, pp. 150~158, 1997.

[5] T. Lane and C.E. Brodly, "An application of machine learning to anomaly detection," 20th NISSC, 1997.

[6] T. F. Lunt, "A survey of intrusion detection techniques," *Computer & Security*, vol. 12, no. 4, June 1993.

[7] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257~286, Febebuary 1989.

[8] C. Warrender, S. Forrest and B. Pearlmutter, "Detecting intrusions using system calls: Alternative data models," *Proc. IEEE Symposium on Security and Privacy*, May 1999.