순서 기반의 커널과 SVM을 사용한 신분위장공격 탐지

서정석° 이영석, 김한성, 차성덕
한국과학기술원 전자전산학과
{jsseo°, yslee, kimhs, cha}@dependable.kaist.ac.kr

Masquerade Detection based on SVM and Sequence-based Kernel Method

Jeongseok Seo° Yeongseok Lee, Han-Sung Kim, Sungdeok Cha
Div. of Computer Science, Dept. of EECS, KAIST and AITrc/IIRTRC/SPIC

요 약

신분위장공격 탐지는 오랫동안 연구되어 왔지만 실제 시스템에 적용되어 사용되기에는 여전이 높은 오탐지율(false alarm)과 낮은 탐지력(detection rate)이 가장 큰 문제였다. 유닉스 시스템에서 신분위장공격을 탐지하기 위하여 사용자의 유닉스 명령어 행위를 프로파일링하고 정상 프로파일링에서 벗어난 권한 도용을 탐지하는 방법을 사용한다. 본 연구에서는 신분위장공격 탐지 시스템의 탐지력을 높이기 위하여 순서 정보를 반영한 SVM 커널 기법을 고찰하고 실험 결과를 정리하였다.

1. 서 론

컴퓨터 시스템에서 다른 사용자의 권한을 도용하여 해당 사용자인 척 자신의 신원을 숨기는 공격자를 "신분위장자(masquerader)"라고 한다. 신분위장공격은 도용한 권한에 따라 데이터 위조, 유출 또는 권한 변경 등의 다양한 공격을 할 수 있다. 신분위장 공격자는 타인의 권한을 이용하여 공격을 수행하므로 침입자를 찾기 위해서는 실제 사용자가 누구인지를 알아내어야 한다. 그러나 먼저 특정 권한이 신분위장공격으로 도용되고 있는지를 탐지해 내는 문제도 매우 어려운 문제이다.

내부자는 보안 시스템을 잘 알고 있어서 다른 사용자의 권한 도용이 용이할 수 있는데, 이러한 내부자에 의한 신분위장공격은 아주 심각한 결과를 나타낼 수 있다. 2004년 CSI/FBI 컴퓨터 범죄에 관한 보고서[1]에 의하면 내부자에 의한 네트워크 오용 피해는 한 해 1060만 달러로 전체 위협 중 4위를 차지하고 있다. 2001년 미국 전 FBI 요원 Robert P. Hanssen[2]의 사건은 내부자에 의한 정보유출의 심각성을 드러난 사례이다.

본 연구에서는 신분위장기법의 탐지를 위하여 사용자의 정상 행위 프로파일을 생성하고, 현재의 행위가 정상 프로파일에서 심각하게 벗어나는 신분위장공격을 탐지하는 연구를 수행하였다. 유닉스 명령어를 이용하여 사용자의 행위 프로파일을 만들기 위해 SVM(Support Vector Machine)을 사용하였다. 이 연구에서는 유닉스 명령어 프로파일링에 적합한 SVM 커널 기법에 대하여 고찰하고 연구 결과를 정리하고자 한다.

2. 유닉스 명령어와 신분위장공격 탐지

Schonlau[3]는 유닉스 명령어 데이터에 신분위장기법 탐지를 위해 6가지 알고리즘을 적용하였다. Maxion[4]과 Kim[5]은 같은 Schonlau의 데이터에 Naive Bayes Classifier 알고리즘과 SVM(RBF) 기법을 사용하여 탐지율을 높였다. 그러나 이 두 알고리즘 또한 실제 시스템에 적용되기에는 여전히 낮은 탐지율과 높은 오탐지율을 가지고 있다.

Kim은 SVM과 RBF 커널을 사용한 신분위장공격 탐지 연구에서 프로파일링을 위해 명령어의 빈도 정보를 사용하였다. 표 1은 유닉스 명령어의 빈도와 순서 정보에 대한 예를 보여준다. 3개의 명령어를 하나의 단위(window size, K)로 사용할 때, <ls, cpp, cd>와 <cpp, cd, ls> 유닉스 명령어 벡터는 빈도 정보를 사용하는 SVM에서는 서로 구분되지 않지만 순서 정보를 사용하는 SVM에서는 서로 다른 입력으로 구분된다. 실제 사용자의 명령어 사용 행위에는 명령어 간의 순서 정보가 중요한 영향을 미친다. 왜냐하면 같은 명령어도 순서에 따라 시스템에 다른 결과를 나타내기 때문이다.

프로파일링을 위해 빈도 정보를 사용하면 벡터의 크기는 전체 feature 수와 같아지게 된다. 반면에 순서 정보를 사용하면 고려할 순서(sequence)의 길이가 벡터의 크기가 된다.

표 1. 빈도 벡터와 순서 벡터(window size = 3)

유닉스 명령어 입력(명령어 스트림) : ls, cpp, cd, ls, make, cd, cd, ···	
빈도 정보를 사용한 feature vector (K=3) <n(cd), n(cpp), n(ls), n(make), n(netscape)>	순서 정보를 사용한 feature vector (K=3)
<1, 1, 1, 0, 0> <1, 1, 1, 0, 0> <1, 0, 1, 1, 0> ···	<ls, cpp, cd> <cpp, cd, ls> <cd, ls, make> ···

다음 장에서는 SVM에 순서 정보를 사용하기 위한 커널 기법들을 소개하고 각 커널 기법들의 실험 결과를 정리한다.

3. SVM and Sequence-based Kernel Method

SVM은 통계 학습 이론을 기반으로 Vapnik에 의해 제안되었으며[6], 그 뒤 많은 분야에 적용되고 있는 기계 학습이론 중 하나이다. 간단한 SVM은 입력 공간(input space)에 있는 분류 데이터들을 마진(margin) 값을 최대로 하는 초평면(hyperplane)을 찾아내어 이진 분류를 수행하게 된다. 이 때 분류기의 성능을 높이기 위한 방법으로 입력 공간의 데이터를 특성 공간(feature space)로 사상(Φ)시키게 된다. 아래 수식과 같이 사상 함수 Φ에 대한 함수 K를 커널(kernel) 함수라고 부른다.

$$K(\vec{x},\vec{x'}) = \Phi(\vec{x}) \cdot \Phi(\vec{x'}) \quad ---- \quad (1)$$

순서 정보를 사용하기 위한 대표적인 기법으로는 가장 간단한 K-gram 커널과 string 커널이 있다. K-gram 커널은 수식 (2)과 같이 K-gram 빈도를 특성(feature)으로 사용하는 방법으로 알파벳(Σ) 입력 스트링을 Σ^k 크기의 특성(feature) 벡터로 사상시킨다.

$$K_n(s,t) = \sum_{u \in \Sigma^n \vec{i}} N(\vec{i},s) \cdot N(\vec{i},t) \quad ---(2)$$

$N(\vec{i},A)$ = Number of exact sequence i in command A

수식 (3)과 같이 K-gram과 RBF 커널의 양의 일차결합(positive linear combination)도 커널 함수이다.

$$K_n(s,t) = \exp\left(-\gamma \left| \sum_{u \in \Sigma^n \vec{i}} N(\vec{i},s) - N(\vec{i},t) \right|^2\right) \quad ---(3)$$

$N(\vec{i},A)$ = Number of exact sequence i in command A

Lodhi[7]는 문서 분류를 위해 K 길이의 연결되지 않는 부분열(subsequence)을 특성 공간에서 사용하는 string 커널을 제안하였다. String 커널도 입력 스트링을 Σ^k 크기의 특성 벡터로 사상시킨다.

$$K_n(s,t) = \sum_{u \in \Sigma^n \vec{i}} \sum_{u=s(\vec{i})} \sum_{u=t(\vec{j})} \lambda^{i_n + j_n - i_1 - j_1 + 2} \quad ---(4)$$

Σ: a finite alphabet (command)

Σ^n: sequences of length n over command Σ

$\vec{i} = (i_1, \ldots, i_n)$: index sequence (sorted)

$s(\vec{i})$: substring operator

$r(\vec{i}) = i_n - i_1 + 1$: range of index sequence

앞의 3가지 커널 입력 s, t는 벡터가 아닌 유닉스 명령어의 문자열이며, 두 문자열 s와 t의 길이는 서로 달라도 상관없다. 그러나 K-gram과 K-gram_RBF 커널은 연결되어 있는 부분열(k-length exact subsequence)을 고려하는 반면에 string 커널은 연결되어 있지 않은 부분열(k-length noncontiguous subsequence)을 고려한다.

4. 실험 및 결과

순서 정보 기반의 커널 성능을 비교 평가하기 위하여 기존 연구들[3,4,5]에서 사용하였던 Schonlau 데이터를

표 2. 실험 결과

	Hits(%)	False Positive(%)	Misses(%)	Cost (1:1)
SVM (K-gram_RBF, K=2)	89.61%	14.86%	10.39%	25.25
SVM (String Kernel, K=2)	97.40%	23.77%	2.60%	26.37
SVM (K-gram, K=2)	96.54%	24.76%	3.46%	28.22
SVM (RBF kernel)	80.09%	9.71%	19.91%	29.62
Bayes 1-Step Markov	69.30%	6.70%	30.70%	37.40
N.Bayes(no Upd.)	66.20%	4.60%	33.80%	38.40
N.Bayes (Updating)	61.50%	1.30%	38.50%	39.80
Hybrid Multi-Step Markov	49.30%	3.20%	50.70%	53.90
IPAM	41.10%	2.70%	58.90%	61.60
Uniqueness	39.40%	1.40%	60.60%	62.00
Sequence Match	36.80%	3.70%	63.20%	66.90
Compression	34.20%	5.00%	65.80%	70.80

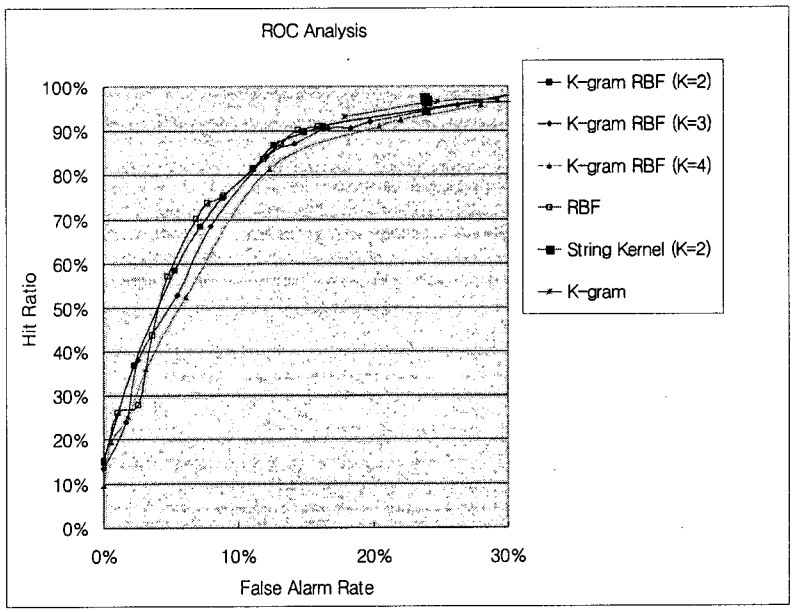

그림 1. ROC Analysis

이용하여 실험하였다. 실험 결과(표 2)를 보면 SVM이 상대적으로 높은 양의 오탐지율(false positive)을 보이지만 상당히 낮은 음의 오탐지율(false negative, misses)을 보인다. 좋은 침입탐지시스템은 낮은 양의 오탐지율과 낮은 음의 오탐지율을 가져야 하는데, 이를 평가하기 위하여 아래와 같이 비용(cost) 함수를 사용한다. 비용이 낮을수록 좋은 탐지 알고리즘을 의미한다.

$$Cost = \alpha \cdot F.N. + \beta \cdot F.P. \quad \text{---} \quad (5)$$

비용 함수 파라미터 α와 β는 탐지 시스템의 적용 도메인에 따라 다양하게 설정된다. NSA, NIS 등의 국가정보기관은 양의 오탐지율(false positive)보다는 상대적으로 낮은 음의 오탐지율(false negative, 공격자 탐지 실패율 최소)가 중요하게 된다. 결과적으로 비용함수는 공격자를 탐지하지 못하였을 때의 비용과 정상 사용자를 오판하는 비용의 중요도에 따라 결정된다. 이 연구의 효용성을 평가하기 위하여 $\alpha = \beta = 1$을 사용하였다.

본 연구에서는 신분위장공격 탐지의 효율을 높이기 위하여 유닉스 명령어의 순서 정보를 행위 프로파일링에 사용하였으며, 그 결과로 탐지기의 성능이 향상되었음을 보여준다. 표 2의 실험 결과와 그림 1의 ROC 분석 차트를 보면 양과 음의 오탐지율을 모두 고려한 비용(cost)을 비교하였을 때 K-gram과 RBF 커널을 일차 결합하여 사용하는 K-gram_RBF 커널이 가장 좋은 성능을 보였으며, 부분열(subsequence)의 길이를 2로 적용하였을 때 가장 좋은 성능을 보인다. String 커널은 높은 오탐지율에 반해 높은 탐지력을 보이므로 오탐지율보다 상대적으로 높은 탐지력을 요구하는 도메인에 - 공격자를 놓쳤을 때 큰 비용을 지불하는 고급 정보기관 - 적용하는 것이 효과적이라는 것을 알 수 있다.

5. 참고 문헌

[1] CSI/FBI, "Computer Security Issues and Trends: 2004 CSI/FBI Computer Crime and Security Survey," Computer Security Institute, 2004.

[2] William H. Webster et.al, "A Review of FBI Security Programs," March 2002.

[3] M. Schonlau et.al, "Computer Intrusion: Detecting Masqueraders," Statistical Science, 16(1) 2001.

[4] Roy A. Maxion and Tahlia N. Townsend, "Masquerade Detection Using Truncated Command Lines," Proc. Int'l Conf. Dependable Systems and Networks(DSN-02), June 2002.

[5] H. Kim and S. Cha, "Efficient Masquerade Detection Using SVM based on Common Frequency in Sliding Windows," IEICE Trans. Info.&Syst., VolE87-D, No.11, Nov. 2004.

[6] V. Vapnik, "Statistical Learning Theory," John Wiley, 1998, NY.

[7] H. Lodhi et.al, "Text Classification using String Kernels," The Journal of Machine Learning Research Vol 2, March 2002.