위험도 분석에 기반한 On-demand 방식의 호스트 침해 증거 자료 수집 방안

*최 윤 호, 박 종 호, 김 상 곤, 서 승 우, **강 유, 최 진 기, 문 호 건, 이 명 수
*서울대학교 전기컴퓨터공학부, **KT
e-mail : *[yuno, jhpark, sangkon.kim]@cnslab.snu.ac.kr, sseo@snu.ac.kr, **[yulguang, jingiya, hkmoon, msrhee]@kt.co.kr

On-demand Evidence Collection of Host Infringement based on the Analysis of Severity levels

*Yoon-Ho Choi, Jong-Ho Park, Sang-kon Kim, **Yu Kang, Jin-Gi Choi, Ho-Gun Moon, Myung-Su Lee
*School of EECS at Seoul National University, **KT

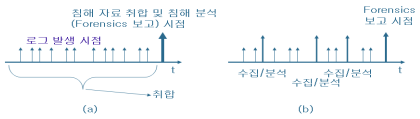
Abstract

Computer Forensics is a research area which finds the malicious users by collecting and analyzing the intrusion or infringement evidence of the computer crime. Many research about Computer Forensics has been done. But those research have focussed on how to collect the evidence after receiving the damage reports from computer users or network administrators. This paper describes about collecting the evidence of good quality at the time of infringement occurrence by the malicious user. By calculating Infringement severity of observable and protective hosts and referring to this value, we collect the evidence at the time of infringement occurrence to minimize the information modification of the evidence. We can reduce also the amount of logs that we use to analyze the infringement and can minimize the loss of the evidence.

I. 서론

컴퓨터 Forensics는 급증하고 다양화 되어 가는 컴퓨터 관련 범죄가 발생할 시, 침입에 대한 전자 증거 자료를 수집하고 분석함으로써 악의적 사용자를 찾아내는 분야로서, 최근 이에 관한 많은 연구가 진행되고 있다. 그러나 기존 Forensics의 경우에는(그림1-(a)), 사건 발생 보고 시점을 기준으로 관련 증거 자료를 수집/보관/분석해 왔다. 이 과정에서 분석 자료가 방대해지고 손상되는 경우가 발생했다. 또한 공격자에 의한 증거자료 훼손 및 변조에 적절히 대응할 수 없었으며, 정상적 사용자의 다른 프로그램 동작 등으로 인한 저장된 증거 자료 훼손의 가능성을 최소화 하지 못했다.

본 논문에서는 사이버 범죄에 적절하게 대응하기 위해 악의적 사용자에 의해 고의적으로 시스템이 침해된 경우, 사건 발생 시점에 기초하여 양질의 증거자료를 효과적으로 수집하기 위한 방안에 대해 제안한다(그림 1-(b)).



[그림 1] 침해사고 보고 비교: (a)기존(취합), (b)제안(수집)

II. 제안된 알고리듬에 대한 기술

공격자에 의한 침입을 침입 시도 및 침해(시스템에 대한 실제 침입이 발생) 관점으로 구분하고, 호스트에 대한 공격이 초기 공격단계(host scan, port scan, vulnerability exploit)에서, 네트워크 침해 경로 설정 단계(infection)를 지나, 최종 공격 단계(예: DoS attack)에 이르는 다단계 공격에 대해 분석한다. 분석 대상 호스트의 침해 발생 시점에서의 각 단계별 침해 위험도를 계산하여 단계별 증거자료 수집을 위한 시점을 판단한다. 위험도 계산 시 잘못된 판단을 최소화하기 위해 침입 탐지 시스템(IDS)인 SNORT와 호스트의 로그(이벤트 로그) 및 보안 설정(configuration) 정보간의 상관관계를 분석한다[1][2]. 한편 대상 호스트의 침해

에 대한 판단 기준 자료 및 증거 자료는 공격 단계별 상태정보로 표현한다. 공격 단계별 severity값에 기초하여 관련 증거자료를 수집하기 위한 알고리즘 및 수집절차는 다음과 같다. 이때, 분석에 사용되어진 각 호스트는 시간 동기화되어 있다고 가정한다.

- (침해) severity: 호스트가 실제 침해되었을 가능성
- v(s, IP): state s에서의 침해된 IP의 severity
- sv(s, ss): state s에서의 ss번째 하위 state에서의 sub-severity
- pre-condition: 호스트 침해가 성공하기 위한 조건
- post-condition: 호스트 침해에 관한 보고
- c_k: 침해 성공가능성과 관련된 조건, (pre- condition, post-condition), (k=1,2,...)
- c_{ssm}: 침해 성공가능성과 관련된 하위 state ss에서의 m번째 조건,(pre-condition, post-condition), (ss, m=1,2,...)
- C(s): state s에서의 침해 판단을 위한 조건 집합, c_k 혹은 c_{ssm} 로 구성 (s:1~5 or HS:1, PS:2, VE:3, D(Dameon의 약자):4, DDoS:5), 으로 구성 원소의 post-condition을 구성하는 분석 자료가 동일한 침해 증거를 지시하는 경우 중요도에 따라 부분집합(sub-severity계산을 위한 조건)으로 구성됨.
- |C(s)|: C(s)의 element 수
- wt(s): state s 에서의 severity계산에 사용되는 조 weighting factor
- wt(s, k or ss): state s에서의 c_k혹은 c_{ss}의 네트워크 관점 혹은 호스트 관점에서의 중요도에 따라 주어지는 조건 weighting factor (k: 1~|C(s)|, ss: 1~C(s)의 부분집합의 수)
- wt(s, ss, m): state s에서의 m개의 조건에 의해 주어지는 ss번째 하위 state에서의 sub-severity 계산을 위한 weighting factor
- Lo_N: 상태별 분석 대상 로그 파일의 수
- Li_N: 분석 대상 로그파일의 라인 수

Step1. 주기적 시간 간격을 두고 단일 호스트로부터 수집된 로그를 각 침해 단계에서 사용되는 정보로 분류/분석(C(s)의 post- condition) 후, 각각의 상태 정보와 관련된 정보를 저장한다.

Step2. 수집된 로그에 대해 시간적 우선순위와 시스템 상태 정보에 기초하여 관련 정보를 분석한다. 그림 2의 알고리즘을 이용하여 각 공격 상태별로 공격에 대한 분석 대상 호스트(IP)의 severity(v(s, IP))를 계산한다.

Step3. Step2에서 계산되어진 severity값을 기준으로, 다음의 분석 대상 호스트들의 침해 상황별로 각 단계의 증거 수집 대상 호스트와 수집되어야 하는 증거자료를 정의한다.

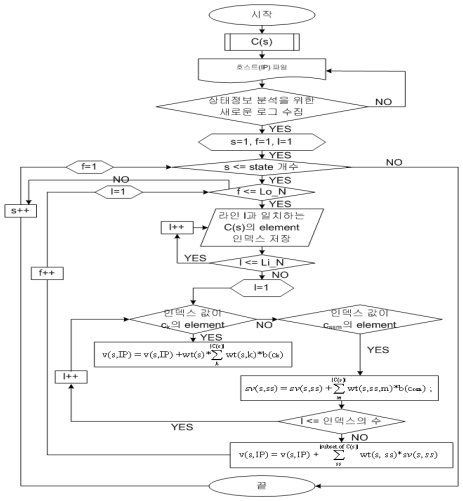
Case 1. 특정 호스트의 침해 상황

Case 2. 특정 호스트 및 주변 감시 대상 호스트의 침해 상황

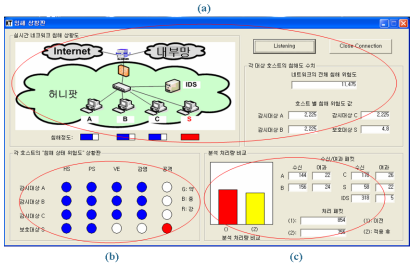
Case 3. 감시 대상 호스트들이 침해당하고 보호 대상 호스트가 공격 받았을 가능성이 높은 상황

Ⅲ. 구현

그림 3-(a)와 (b)에 보여 진 바와 같이, DDoS공격의 진행과 동시에 감시 대상 호스트(A, B, C)의 침해(흰색->녹색->적색) 사실과 최종 공격이 보호 대상 호스트(S)에 가해진 것(적색)을 확인할 수 있다. 이러한 각 호스트의 침해 위험도 값을 바탕으로, 침해 시점에 기초하여 각 침해 상태별 Forensics을 위한 증거자료를 수집할



[그림 2] step 1과 step 2를 통한 단계별 severity계산을 위한 알고리듬 flow chart, O(Li_N) (Lo_N<<Li_N && 인덱스 수<<Li_N)



[그림 3] Prototype 소프트웨어 구성도 및 동작 화면(서버): (Severity Management(SM)-Server, SMS라 함.)

수 있었다. 또한, 그림 3-(c)에 보여 진 바와 같이, T=30sec(10sec)인 경우, 기존 방안 적용 시, 전체 199213byte의 로그를, 제안된 방안 적용 시, 80719byte (1681byte/T)(T=10sec: 100624byte(708byte/T))의 로그를 수집하여, 60%(T=10sec: 49.5%)의 분석에 사용되어지는 로그 량 감소 효과를 가져 올 수 있었다.

Ⅳ. 결론

이 논문에서는 다단계 공격에 대해 호스트의 실제 침해당한 시점에 기초하여 침해 상황을 감지하고 관련 증거 자료를 수집하기 위한 방안을 제안하였다. 이는 침해 시점에 기초하여 자료를 분석하고 증거 자료를 수집함으로써 증거자료의 손상을 최소화한다.

참고문헌

[1] HervéDebar, Andreas Wespi, “Aggregation and Correlation of Intrusion Detection Alerts”, in proceedings of RAID, 2001.

[2] Frederic Cuppens, Alexandre Miege, “Alert Correlation in a Cooperative Intrusion Detection Framework”, in proceedings of IEEE S&P, 2002.