행위 기반 악성코드 탐지 · 차단시스템 개발

김성우[1], 신재인[2], 방영환[3]

Development of Behavior-based Malicious code etection · Blocking System

Sung-Woo Kim[1], Jae-In Shin[2], Young-Hwan Bang[3]

요 약

인터넷 인프라의 급속한 발전과 인터넷 보급율의 확대에 따라 사용자 PC의 보안을 위협하는 악성 프로그램이 갈수록 지능화, 다양화되고 있고, 악성 프로그램에 의한 피해는 나날이 커지고 있다. 또한 컴퓨터 성능의 발전 속도가 발라짐에 따라 피해 상황도 빨라지고 있다. 따라서 최근 보안 프로그램의 특징은 진화된 악성프로그램을 제어하기 위하여 실시간 감시, 바이러스, 안티스파이웨어가 결합된 프로그램이 출시되고 있다. 그러나, 지능적으로 변모하는 변종 유해 프로그램에 대해 실시간으로 대응할 수 없다는 문제점이 있다. 이에 본 논문에서는 전통적인 signature나 heuristic기법이 아닌 주요 운영체제 함수를 실시간 감시함으로 프로세스의 행동 sequence를 추적하여 행동조합 시나리오 개념에 의한 매칭으로 악성 프로그램인지 판별하며 알려지지 않은 악성 프로그램 또는 변종까지 차단할 수 있으며 signature가 변하는 문제점을 해결하고 zero-day 공격까지 방어할 수 있는 방법을 제시하고자 한다.

핵심어 : 행위기반 탐지, 보안공학, 위험분석, 복합시나리오

Abstract

The rapid development of Internet infrastructure and the expansion of Internet penetration, depending on your PC malicious programs that threaten the security of increasingly intelligent, diversified, and the damage caused by a malicious program that is growing day by day. In addition, the performance of your computer faster, depending on the speed of the development has speeded up the damage situation. Therefore, the characteristics of recent security program to control the evolution of real-time monitoring of malware, viruses, combined with anti-spyware program has been launched. However, a variant of transforming harmful programs that intelligently about not being able to respond in real time there is a problem. In this paper, the traditional signature or heuristic techniques rather than a major operating system function, real-time monitoring by the process's behavior sequence by tracking the behavior combination scenario concept by matching the malicious program, determine whether and unknown malicious programs or variants to be blocked and the signature is solve problems and changes to zero-day attack is to propose a way to defend.

Keywords : Behavior Detection Tech, Security Engineering, Risk Analysis, Complex Scenarios

접수일(2012년02월07일), 심사의뢰일(2012년02월08일), 심사완료일(1차:2012년02월28일, 2차:2012년03월04일)

게재일(2012년04월30일)

[1]306-791 서울시 강남구 수서동 725 수서타워 201호, 아펙스씨앤에스 대표이사.
email: sw.kim@apexcns.com

[2]100-100 대전광역시 서구 대덕대로 168번길 74, 스컴씨엔에스 연구원.
email: mfshaaa@gmaiil.com

[3](교신저자) 331-825 충청남도 천안시 서북구 입장면 홍천리 35-3, 한국생산기술연구원 책임연구원.
email: bangyh@kitech.re.kr

* 본 논문은 지식경제부 2011년 "개인 및 기업 맞춤형 서비스를 위한 개방형 모바일 클라우드 용 통합개발환경 및 이기종 단말-서버 간 협업 기술 개발" 사업의 연구지원.

1. 서론

인터넷 인프라의 급속한 발전과 인터넷 보급율의 확대에 따라 사용자 PC의 보안을 위협하는 악성 프로그램이 갈수록 지능화, 다양화되고 있고, 악성 프로그램에 의한 피해는 나날이 커지고 있다. 고성능 컴퓨터가 봇넷의 특정 좀비 컴퓨터로 이용되거나 웜에 감염되어 또 다른 전염 대상 컴퓨터를 찾을 때 컴퓨터 성능의 발전 속도가 발라짐에 따라 피해 상황도 빨라지고 있다[1][2].

악성 프로그램은 그 종류에 따라서 다양한 형태가 존재하지만, 다른 프로그램 또는 운영 체제에 접근하여 코드를 변경시키거나 정보를 추출하는 동작, 비정상적인 네트워크 패킷을 송수신하는 동작 또는 보안 프로그램으로부터 자신의 존재를 숨기기 위한 은닉 행위와 같은 일반적인 프로그램과는 다른 이상 행동을 수행한다는 공통적인 특성을 가지고 있다. 또한, 바이러스, 백도어, 루트킷, 트로이 목마와 같은 악성 코드가 초창기에는 각각 개별적으로 움직이는 반면 최근에는 이들이 복합적으로 발생하여 이를 제어하는데 많은 어려움이 있다.

특히 인터넷에 돌아다니는 악성코드는 대부분이 오픈 소스 코드로 공개되어 있어 누구라도 악성 코드 조작하여 변종 악성 코드를 배포할 수 있다. 이로 인해 보안 취약점이 발생한 이후 채 하루가 되기 전에 악성코드에 의한 공격이 들어오는 제로 데이 공격이 현실화 되는 문제점이 있다.

최근 보안 프로그램의 특징은 진화된 악성프로그램을 제어하기 위하여 실시간 감시, 바이러스, 안티스파이웨어가 결합된 프로그램이 출시되고 있다. 이들 제품의 특징은 백신 DB를 근간으로 실시간 탐지를 수행하거나, 일부 행동기반 기능이 첨부된 형태이다[4].

그러나, 종래의 악의적인 프로그램을 취득하여 코드를 분석하여 악의적인 프로그램을 제거할 수 있는 패턴 시그니쳐를 만들어야 악의적인 행동을 막을 수 있는 시그니쳐 방식과, 기존 유해 프로그램의 코드를 분석하여 향후 발생할 수 있는 유해 프로그램의 유입 및 행동을 막고자 나온 heuristic기술은 유사 코드패턴 을 가진 유해프로그램에 대해서는 대처가 가능하나 신규로 발생하는 유해 프로그램 및 지능적으로 변모하는 변종 유해 프로그램에 대해 실시간으로 대응 할 수 없다는 문제점이 있다[3].

본 논문에서는 전통적인 signature나 heuristic기법이 아닌 주요 운영체제 함수를 실시간 감시함으로 프로세스의 행동 sequence를 추적하여 행동조합 시나리오 개념에 의한 매칭으로 악성 프로그램인지 판별하며 알려지지 않은 악성 프로그램 또는 변종까지 차단할 수 있으며 signature가 변하는 문제점을 해결하고 zero-day 공격까지 방어할 수 있는 방법을 제시하고자 한다.

2. 프로세스 행동감시/복합시나리오 매칭

2.1 프로세스와 행동감시

행동기반 유해 프로그램 검출/차단 시스템은 기존의 signature기법이나 heuristic기법이 아닌 프

로세스의 행동을 감시하여 해당 프로세스가 악성프로그램인지 결정하여 차단하는 방법으로 이미 알려진 악성 프로그램과 신종/변종악성 프로그램도 signature없이 검출/차단이 가능하다.

pc에서 작동하는 모든 프로세스는 각자 행하는 목적은 다르지만, 그 행위들을 단위로 쪼개어 보면 운영체제가 지원해 주는 기능들을 조합하여 목적을 이루게 된다. 프로세스가 행하는 파일/네트워크 행위들은 그 내용들이 무엇이거나 발생 패턴이 아무리 변화할지라도 운영체제가 지원하는 파일/네트워크크 함수를 사용하지 않고는 그 기능을 수행할 수 없다. 아무리 복잡하거나 많은 일을 하는 프로세스라도 그 프로세스가 사용하는 운영체제 주요함수의 갯수는 유한하다 라는 결론에 도달한다 이는 3차원 공간에서 무한한 좌표지점이 존재하지만 3차원에 모두 포함된다는 개념과 동일하다[6].

그럼으로 운영체제에서 지원하는 이러한 주요기능 함수들을 감시하여 모든 프로세스가 실시간으로 행하는 복잡한 행동들을 행하는 순간을 정확히 포착하며 그 행동들을 행한 순서와 시간 등을 알아낼 수 있게 되어 순서정보를 갖는 행동 sequence가 도출되며 이는 행동기반으로 프로세스를 감시할 수 있는 근거가 된다.

운영체제가 지원해 주는 행위들 중에는 보편적이고 위험하지 않은 행동부터 거의 발생하지 않는 위험한 행동들이 존재하게 된다[7].

시스템의 주요한 행동목록은 외부네트워크 접속, 다중ip에 패킷전송행위, IP/MAC/ARP변조패킷 발송, 파일오픈, 파일오픈/생성, 시스템 IDT hook, 서비스 생성/시작/오픈, 물리 메모리 접근, 프로세스 실행, 타 프로세스 접근, 타 프로세스 메모리 공간에 기록, 운영체계 주요 함수 테이블 침범(SDT hook), 자신을 은닉하는 프로세스, 파일을 은닉, 서비스 은닉, 타 프로세스에 원격작업 실행, 레지스트리 자동시작 등록, 키보드 해킹시도, 레지스트리 은닉, 이름 없는 프로세스, 부모없는 프로세스, 실행파일 생성, 실행파일 쓰기모드 open, 디바이스 드라이버 로드, 타 프로세스 강제종료 등으로서 일반적인 행위부터 위험한 행위까지 존재한다.

다음은 프로세스가 행하는 행동 sequence의 예시이다.

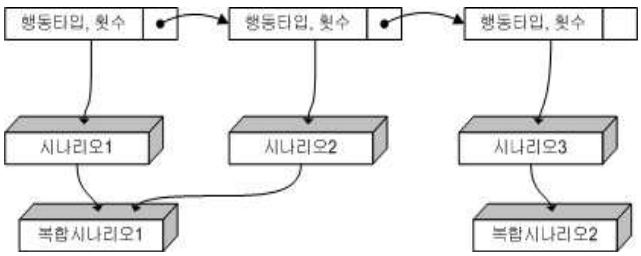
- p1(유해 프로세스) : {외부네트워크 접속, 실행파일 생성, 실행파일 자동시작 등록, 실행파일 시작, 실행프로세스 숨김}
  - 외부에서 실행파일을 다운받아 레지스트리에 자동시작으로 등록후 실행하여 검출되지 않게 은닉해 주는 웜 바이러스
- p2(유해 프로세스) : {외부네트워크 접속, 실행파일 2개 생성, 서비스 등록, 실행파일 자동시작 등록, 실행파일 시작, 실행프로세스 숨김}
  - 외부에서 rootkit실행파일과 driver를 다운받아 driver는 서비스로 등록하고 실행파일을 레지스트리에 자동시작으로 등록후 실행/은닉하는 rootkit

2.2 행동 sequence와 복합 시나리오 매칭

행동기반으로 임의의 프로세스가 정상 프로세스인지 유해 프로세스인지 구분하는데 있어서 운영체제가 지원하는 위의 행동들은 단일 행동으로는 전혀 위험하지 않은 것도 존재하며 단일 행동으로도 매우 위험한 행동들이 존재한다. 만일 각각의 행동들을 행하는 프로세스를 차단하게 된다면 운영체제 자신도 차단되며 정상 프로그램도 차단될 수 있는 대규모 오탐이 발생하게 된다. 이에 필요한 것이 시나리오 개념인데 시나리오라는 것은 행동 sequence를 의미 한다 연속된 서로 다른 행동타입을 제시하여 그것에 모두 매칭 되는 것을 유해 프로그램으로 간주하는 기법이다. 시나리오 내부에는 1개 이상의 행동타입과 행동횟수가 들어가며 이러한 조건이 많아질수록 정상 프로그램은 탐지되지 않을 확률이 높게 되는데 시나리오 개념을 쓰는 원인이 된다. 또한 서로 다른 행동들이 조합시에 위험도가 높아진다는 특성이 존재 한다 "외부네트워크 접속"행동과 "실행파일 생성"행동이 따로 존재 할때는 위험도가 낮으나 함께 있을 때는 위험도가 높아진다. {"외부네트워크 접속", "실행파일 생성"}, 이는 외부에서 악성프로그램을 다운받는 행위로 간주될 수 있다[8].

1차 개발결과와 유해 프로그램 테스트 결과를 거치면서 기존의 시나리오 방식에서 더 진보한 복합시나리오 방식을 사용하게 되었는데 이는 n개의 시나리오를 하부데이터로 가지는 방식으로 기존의 시나리오 개념에서는 시나리오에 존재하는 모든 행동이 매칭시 해당 프로세스는 그 시나리오에 의하여 차단이 되지만, 복합시나리오는 하부의 n개의 시나리오에 속하는 모든 행동이 발생되어 매칭시 1개의 복합시나리오가 매칭되고 차단이 발생하는 것이다. 복합시나리오는 1개에서 n개까지 하부 시나리오를 가지며 1개의 시나리오만을 가지는 복합시나리오는 기존의 단일 시나리오 방식으로 작동하게 된다. 다음은 단일 시나리오 2개와 복합시나리오 1개에 대한 예이다.

- S1(시나리오) : {외부네트워크 접속, 실행파일 생성, 서비스 등록, 레지스트리 자동시작 등록, 실행파일 실행, 프로세스 은닉}
- S2(시나리오) : {타 프로세스 강제종료, 파일생성, 다중ip에 패킷전송}
- 복합시나리오1 : {S1, S2}



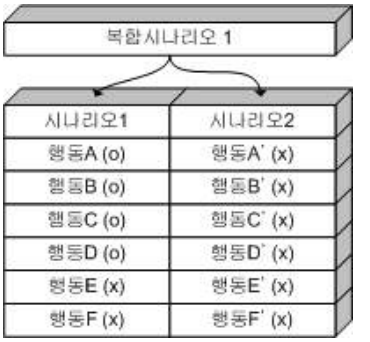
[그림 1] 복합시나리오 프로세스

[Fig. 1] Complex Scenarios Process

복합 시나리오 n개가 pc의 프로세스 감시모듈에 설정되면 감시모듈은 프로세스들이 행동발생시 아주 빠른 비교/매칭을 위하여 Hash 데이터 구조와 dynamic programming기법을 사용하는데 hash의 key값은 행동타입과 행동횟수를 사용하여 해당 행동을 포함하고 있는 시나리오와 또 그

시나리오를 포함하는 복합시나리오를 빠르게 검색해 낼수 있다.

프로세스의 행동이 발생 시 즉시 해당 행동을 가지고 있는 시나리오가 있는지 검색해야 하기 때문에 데이터 구조는 행동->시나리오->복합 시나리오의 역순으로 구조화 되어 있어서 최상단으로 빠르게 접근할 수 있게 되며 아래의 그림과 같이 발생한 행동에 대해서 O표시로 행동이 발생 되었음를 기록하게 된다. 행동이 발생 시 마다 자신을 O표시로 하고 자신이 속한 시나리오에서 아직 발생이 안된 행동이 존재하지 않으면 그 시나리오에 속한 행동은 모두 발생하였고 다음 시나리오의 매칭을 시작하게 된다. 이런 방식으로 복합시나리오 1개에 속한 여러 개의 시나리오가 행동발생시 모두 매칭이 발생시 해당 프로세스는 차단이 발생하게 된다. 복합시나리오에 있는 시나리오1과 시나리오2는 순차적으로 매칭이 된다.



[그림 2] 복합시나리오 매칭

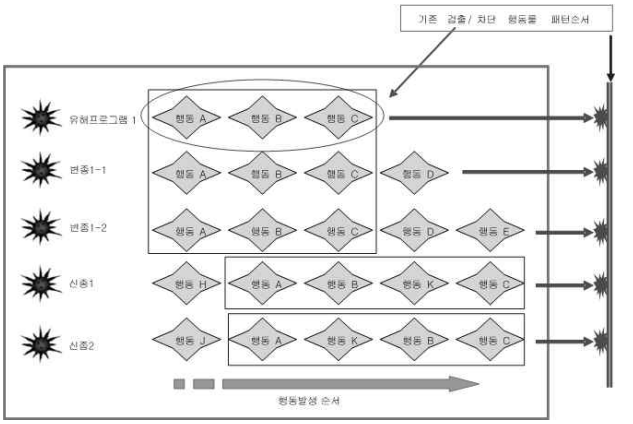
[Fig. 2] Matching Complex Scenarios

2.3 알려지지 않은 유해 프로그램과 변종

복합시나리오에 의한 프로세스 행동매칭 방식은 기존의 알려진 유해 프로그램과 그것의 변종 그리고 아직 알려지지 않은 유해 프로그램까지 검출/차단할 수 있다 이는 변종 프로그램의 경우 기존 유해 프로그램과 거의 행동이 유사하며 알려지지 않은 신종 유해 프로그램도 기존에 존재하는 유해 프로그램과 비교할 때 외형적인 signature나 heuristic의 코드분석 패턴은 다를지라도 행동 영역에서는 유사하기 때문이다. 다음은 행동매칭시의 변종과 신종에 대한 매칭도이다 복합시나리오의 하부에 있는 각각의 시나리오가 매칭시 각 행동사이에는 더미행동개념이 존재한다. 아래의 그림은 단일 시나리오(행동A, 행동B, 행동C)에 의한 매칭이미지이며 그 예를 보여준다.

신종1 유해 프로그램의 경우 {H, A,B,K,C}의 순으로 행동을 행하였는데 단일 시나리오 = {A,B,C}와 매칭 시 행동A와 행동B사이에 행동K가 존재한다. 더미 이상행동은 시나리오가 가지고 있는 하부 행동들 사이에 다른 어떤 행동이 와도 무시한다는 개념으로서 변종이나 알려지지 않은 신종 유해 프로그램이 작동시 이미 설정된 복합 시나리오 정책으로 모두 차단하게 된다. 변종1/2는 기본과 약간 다른 추가적인 행동을 하는 것으로서 기존의 시나리오로 동일하게 탐지/차단이

가능하다.



[그림 3] 단일 시나리오에 의한 매칭

[Fig. 3] Matching by a single scenario

비교적 행동내역의 변화폭이 큰 신종1/2는 최초의 행동이 기존의 유해 프로그램과 다르며 중간에 다른 행동이 발생할지라도 더미행동 개념에 의하여 매칭이 성공하게 되었다 이로인해 기존에 이미 설정된 정책에 의하여 변종과 알려지지 않은 유해 프로그램에 의한 zero-day공격을 방어할 수 있다.

2.4 Agent기반 작동방식과 네트워크 행동과 프로세스 행동의 동일한 취급

pc에서 작동하는 프로세스의 행동을 감시하기 위해서는 반드시 pc에서 Agent방식으로 작동하게 되는데 Agent는 중앙의 서버프로그램에 네트워크 접속을 하여 전체망에 설치된 pc들의 Agent들로부터 데이터를 수집가능하며 복합시나리오 정책도 확산시킬 수 있다. Agent방식의 프로세스감시로 인하여 유해 프로그램이 발생 시 프로세스의 상세정보(이름, 크기, 경로)와 유해 프로그램 본체를 획득할 수 있으며 유해 행위를 행한 pc가 어떤 것인지 그 안의 유해 프로그램을 정확히 알아낼 수 있는 장점이 있다.

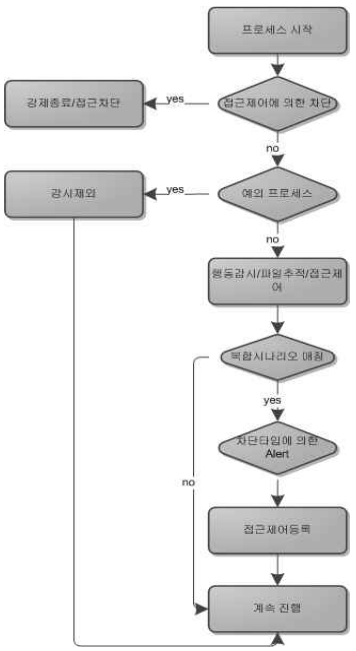
기존의 네트워크 공격을 방어하기 위한 네트워크 장비들은 해당 행위를 한 실제 pc와 프로세스 정보를 알수 없기 때문에 프로세스에 제재를 가하거나 실제 pc정보를 관리자에게 통보하여 후속조치를 취할 수 없다. 이는 대학교 망 내부에서 어떤 pc가 IP/MAC 변조 패킷을 과다를 발생하여 망에 부하가 걸릴 때 실제 pc를 알지 못하여 조치를 취하지 못하는 예와 같다. 행동감시를 수행하는 Agent는 IP/MAC변조 패킷이 발생시 즉시 pc정보와 프로세스 정보까지 상세하게 알아내고 차단가능하다.

기존의 백신 프로그램이나 네트워크 보안장비들은 서로 파일이나 프로세스 행동과 네트워크 행

동을 따로 감시하는 2개의 영역으로 나뉘어져 있었으나 행동기반 감시 Agent는 이 두 행동을 통합하여 행동 sequence를 검출하기 때문에 유해 프로그램을 더 잘 검출할 수 있는 장점을 가지고 있다. 외부에서 실행파일을 다운로드 받아 pc에 설치하는 악성 프로그램이 작동 시 네트워크 장비는 어떤 위험도 알아차릴 수 없는데 이는 데이터 전송량도 적으며 외부접속 폭주현상도 일어나지 않기 때문이다. 그러나 {외부접속 시도, 실행파일 생성, 서비스 등록, 레지스트리 자동시작 등록, 실행파일 생성}이라는 네트워크와 파일행동이 혼합된 방식의 행동 sequence를 볼때는 외부접속시도가 있다는 이유로 현재 프로세스가 실행파일을 해커로 부터 다운받아서 설치하고 있다는 중요한 단서가 될 수 있다.

네트워크 망내의 pc에서 작동하는 Agent들은 중앙의 서버에 접속하여 실시간으로 프로세스들의 행동통계를 전송하여 서버 프로그램은 망내의 모든 pc의 프로세스들의 네트워크 행동 내역을 추적할 수 있다. 네트워크 장비에 의해 의심받지 않는 적은 트래픽을 발생하지만 특정 ip주소에 패킷을 전송하는 다수의 pc와 그 행위를 행한 프로세스(DDoS공격)들을 검출, 주기적으로 해커에 접속하는 네트워크 행동흔적, 특정 ip주소에 패킷을 송/수신한 프로세스를 검색하여 국가기관이나 주요시설에 대한 트래픽을 유발한 프로세스가 있는지 검사, 망내에서 서로 다른 pc간 서로 패킷을 송/수신하는 봇 프로그램과 봇에 명령을 내리는 C&C 프로그램을 검출할 수 있다.

2.5 행동기반 유해 프로그램 작동 구성 및 설명



[그림 4] 행동기반 유해 프로그램 프로세스

[Fig. 4] Behavior-based unwanted application process

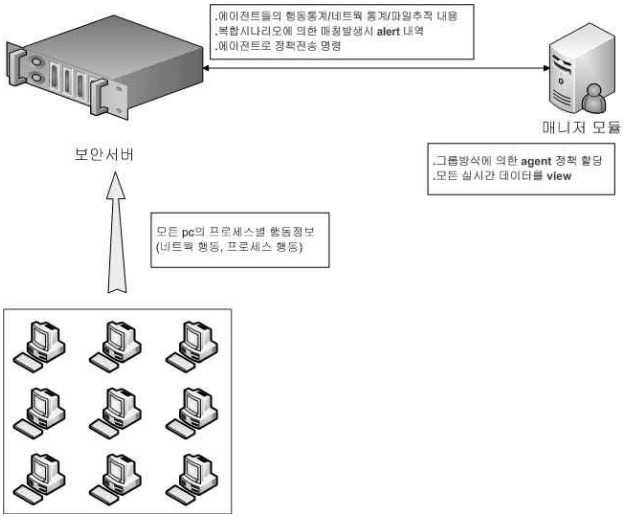
위의 순서도는 행동기반 유해 프로그램 감시/차단 시스템의 주요부분만에 대한 것이며 접근제어에 관련된 내용은 나중에 설명하도록 하겠다. 최초에 프로세스가 시스템에서 실행되는 순간 감시모듈은 프로세스를 감시목록에 둘 것인지 말 것인지 예외 프로세스 목록과의 비교에서 결정하게 된다. 이는 성능증대를 위하여 주요 운영체제파일이나 잘 알려진 프로세스들은 감시하지 않도록 하는 부분이다. 예외 프로세스 목록과 비교시에 프로세스 이름으로 비교하면 운영체제 프로세스로 위장하여 작동하는 악성 프로그램을 감시하지 않게 됨으로 예외 프로세스 목록은 실행파일의 checksum을 미리 저장하여 해당 checksum과 같은 것들은 예외를 허용하게 된다. 감시가 예외가 되면 복합시나리오 매칭도 하지 않게 되며 차단도 하지 않게 된다.

파일추적부분은 해당 프로세스가 생성하거나 기록하거나 실행한 모든 파일에 대한 목록을 통계데이터로 생성해 낸다. 파일확장자는 주요한 “.sys" ".dll" ".exe"만 추적하여 불필요하게 많이 생성되지 않도록 한다. 파일 추적을 하는 이유는 악성 프로그램이 가동시 자신이 생성한 또 다른 dll/exe/sys파일이 무엇인지 확인하여 나중에 악성 프로그램 정보 구축에 활용되며 추가로 악성 프로그램이 다시 그 파일에 접근하지 못하도록 하는 기능을 지원한다. 복합시나리오 매칭 이후에 접근제어 등록을 할때 파일추적에서 발견된 내용을 등록한다. 이것은 복합시나리오에 의하여 매칭된 유해 프로그램이 실행하거나 접근한 exe/sys/dll에 대해서 .sys파일이나 .dll파일은 누구도 쓰거나 읽지 못하도록 하며 자식 프로세스를 생성하지 못하도록 .exe를 등록하게 된다. 이렇게 등록된 내용은 실행파일의 경우 순서도의 맨 처음에 접근제어에 의한 차단에서 접근제어목록과 같은 실행파일이 실행되는 것을 금지하여 유해 프로그램이 자동실행으로 등록한 자식 유해 프로그램이 실행되지 못하도록 하며 유해 프로그램이 다른 프로세스에 injection되어 돌아가도록 설정한 dll이나 sys등도 로딩이 되지 못하도록 “행동감시/파일추적/접근제어” 부분에서 차단한다.

각각의 복합시나리오는 자신과 매칭되는 프로세스에 대해서 어떻게 차단을 적용할 것인지 차단 행동정보를 가지고 있는데, "Alertonly", "강제종료", "해당 프로세스 네트워크 영구차단", "특정 주기만큼만 해당 프로세스 네트워크 차단"등이 존재한다. Alertonly는 매칭이 되어도 어떠한 제재도 가하지 않고 매칭이 되었다는 정보와 함께 행동을 어떻게 행하였는지의 상세정보를 발생시킨다. 강제종료는 유해 프로그램을 그순간 강제로 종료하게 된다.

3. 시스템 테스트/운영 결과

본 시스템이 작동하는 데는 기본적으로 3가지 프로그램 모듈이 존재하는데, 각 pc에 설치되어 모든 프로세스의 행동을 감시하며 복합시나리오 정책과 예외 프로세스 정책을 받아 차단을 수행하는 agent프로그램과 중앙에 서버 프로그램으로 자신에 접속한 모든 agent들의 실시간 행동통계와 차단내역을 수집하며 복합시나리오 정책을 전송하는 보안서버 프로그램, 모든 pc의 상태정보와 행동통계/파일추적/차단alert 현황을 보여주며 ip그룹별로 명령을 내릴 수 있는 매니저 프로그램이다.



[그림 5] 시스템 개념도

[Fig. 5] System concept

매니저 프로그램은 모든 pc의 에이전트를 ip대역에 의한 그룹별로 설정하여 그룹단위로 정책/명령을 보내기 때문에 최초 가동환경은 테스트pc를 그룹으로 만들고 복합시나리오 정책과 예외 프로세스 정책을 미리 설정 및 네트웍 접속까지 완료해 놓은 상태에서 테스트를 수행한다. 설정한 복합시나리오의 이름은 TS-3이고 하부에 2개의 시나리오를 가지고 있고 각각 W1.2와 W1.3이다. 각각의 주요 행동내역은 W1.2는 {실행파일생성, 실행파일 실행) W1.3는 자신을 은닉하는 행위이다. 상세사항은 다음과 같다.

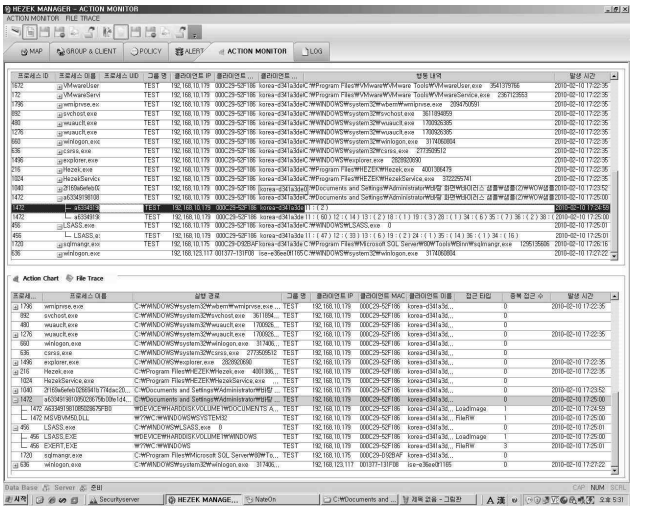
[그림 6] 프로그램 인터페이스

[Fig. 6] Program interface

[그림 7] 프로그램 인터페이스

[Fig. 7] Program interface

테스트 과정은 vmware에서 가상os pc 복합시나리오를 설정하지 않고 없이 먼저 유해 프로그램을 한번 가동한다.



[그림 8] 유해프로그램 가동 인터페이스

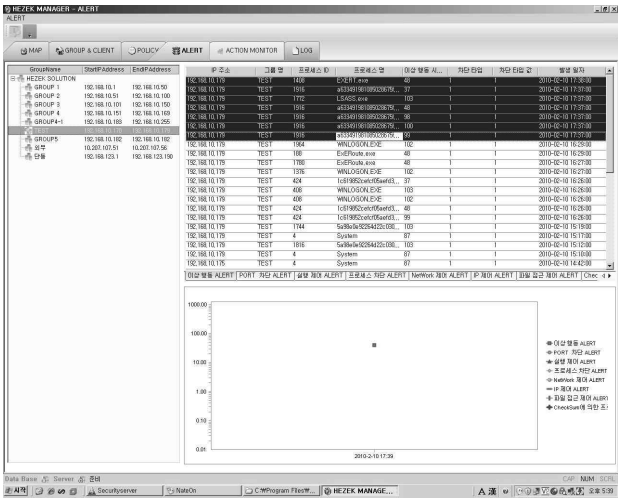
[Fig. 8] Operation unwanted interface program

해당 가상os는 유해 프로그램을 한번 가동하면 매번 폐기하고 원본의 복사본을 매번 생성한다. 매니저의 행동내역에 해당 유해 프로그램의 행동내역이 올라오면 그것을 근거로 최초의 복합시나리오를 생성한다. 다시 복사본 가상os을 실행하여 동일하게 agent를 가동 후 방금 생성한 복합시나리오를 가상os가 속한 그룹에 할당하면 즉시 agent로 명령이 전송된다. 유해 프로그램을 다시

실행시 agent는 복합시나리오 매칭에 성공하고 발송한 차단 alert결과는 중앙의 서버로 전송된다. 이러한 결과를 매니저 프로그램에서 확인하면 테스트 절차가 완성된다.

사용된 유해 프로그램은 WorlfofWarcraft 온라인게임에서 계정을 탈취하는 행위를 하는 a6334919810850286751fb00fe1d4ed1a.exe이며 행동내역은 다음과 같다. 복합시나리오의 추가는 해당 프로세스의 행동내역을 선택하여 추가하면 자동으로 만들어지게 된다. 현재 행동내역은 1초에 한 번씩 발생하게 되는데 만일 유해 프로그램이 3초동안 가동되어 3줄의 행동내역이 올라오면 모두 선택하여 복합시나리오로 등록시 하부에 3개의 시나리오를 갖는 복합시나리오가 추가된다. 위의 이미지의 상단부분은 행동내역이고 하단부분은 파일접근추적내용이다. 본 유해 프로그램은 LSASS.exe와 EXERT.exe라는 자식 프로세스를 실행하여 작동하게 된다. LSASS.exe은 윈도우 운영체제와 동일한 이름의 프로세스이지만 checksum값이 운영체제와 다르기 때문에 예외로 허용되지 않아서 행동통계 내역이 올라왔다. 예외가 되지 않은 프로세스는 행동통계에 모두 올라오며 행동감시/복합시나리오 매칭이 적용되며 차단이 적용될 수 있다.

행동통계 내역을 근거로 위에서 설명한 TS-3복합시나리오(ID : 99) 정책을 수립하여 하부에 2개의 시나리오를 갖도록 추가하였다. 복합시나리오의 차단타입은 강제종료로 하였으며 다음은 차단된 결과물이다.



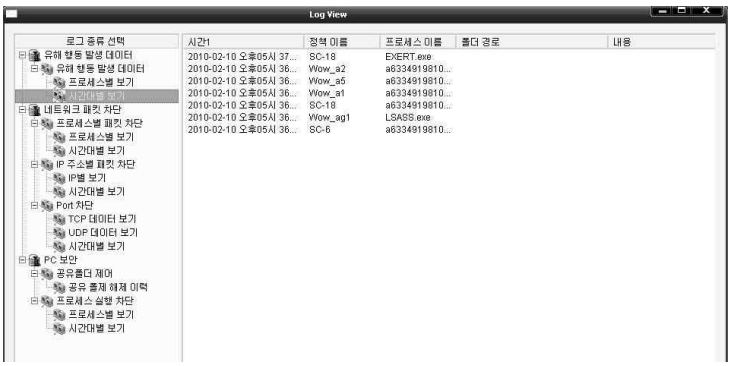
[그림 9] 유해프로그램 차단 화면

[Fig. 9] Block unwanted program screen

실행된 프로세스는 총 3개이지만 차단alert의 갯수는 7개가 발생하였다. a6334919810~1.exe는 5개의 복합시나리오(ID : 100, 99, 98, 48, 37)에 매칭이 되었고 LSASS.exe는 ID:103번인 복합시나리오에 차단되었고 EXERT.exe는 48번 복합시나리오에 의해 차단되었다. a6334919810~1.exe을 차단하기 위하여 새로이 추가된 복합시나리오는 TS-3(ID : 99)였지만 기존에 이미 설정된 다른 유해 프로

그램을 차단하기 위한 복합시나리오에 의하여 다중 alert가 발생하였다. 나머지 2개의 자식 프로세스도 기존에 이미 설정된 복합시나리오에 의하여 차단이 되었다. 이러한 차단은 신종 유해 프로그램이 ZERO-DAY공격이 발생시 복합시나리오를 추가하지 않고 기존의 정책으로 추가 자동차단 되는 효과를 낳는다.

행동기반 유해 프로그램 검출/차단 시스템에 최초에 설정되는 복합시나리는 기존에 알려진 유해 프로그램을 토대로 수립되며 기존에 설정한 정책으로도 잡히지 않는 유해 프로그램은 행동통계를 통하여 CERT-TEAM에 의하여 모니터링 및 의심 프로세스로 간주시 새로운 복합시나리오가 추가된다. 다음은 AGENT에서 발생한 차단로그와 화면으로 시나리오 차단로그와 실시간 alert창의 내용이다.



[그림 10] 차단로그 화면

[Fig. 10] block log screen

[그림 11] 실시간 alert 창 화면

[Fig. 11] Real-time alert window screen

4. 결론

악성 프로그램은 그 종류에 따라서 다양한 형태가 존재하지만, 다른 프로그램 또는 운영 체제에 접근하여 코드를 변경시키거나 정보를 추출하는 동작, 비정상적인 네트워크 패킷을 송수신하는 동작 또는 보안 프로그램으로부터 자신의 존재를 숨기기 위한 은닉 행위와 같은 일반적인 프로그램과는 다른 이상 행동을 수행한다는 공통적 인 특성을 가지고 있다. 기존의 바이러스 백신이나 안티스파이웨어 등의 보안 프로그램은 이미 알려진 형태의 악성 프로그램에 대한 정보(예를 들면, 이진 코드 중의 특정 부분에 대한 패턴 정보)를 기초로 해당 악성 프로그램을 검출하거나 그 실행을 차단하는 방식으로 작동 한다[9]. 그러나, 이와 같은 종래의 보안 프로그램은 특정 악성 프로그램에 대한 이진 코드를 확인하여 패턴 정보를 구성해야만 악성 프로그램의 검출 및 차단이 가능하기 때문에, 그 패턴이 등록되어 있지 않거나 알려지지 않은 악성 프로그램에 대해서는 전혀 대응할 수 없다는 문제점이 있다[10]. 그러므로 악성 프로그램이 수행하는 이상 행동의 조합과 각 이상 행동의 발생 유형에 따라 설정된 유해 프로그램 차단 조건을 기준으로 유해 프로그램을 감지하여 차단함으로써, 알려지지 않은 다양한 유해 프로그램에 대하여 광범위한 보안을 제공하기 위하여 필요한 기술이다.

참고문헌

[1] Vinod Ganapathy, Sanjit A.Seshia, "Automatic Discovery of Api-Level Exploits", Icse 05, 2005.

[2] Birdman, :The Evolution of Wndows Spyware Techniques", HIT2005, July 2005.

[3] A.Sung, J.Zu, P.Chavez, and S.Mukkamala, :Static Analyzer for Vicious Executavles(SAVE)", 20th Annual Computer Security Applications Conference, pp.326-334, December 2004

[4] C.Cortes and V.Vapnid, :Support-Vector Networks, In Machine Learming", pp.273-297, 1995

[5] http://www.facebook.com/note.php?note_id=157088480974999

[6] Bontchev, V. "Macro Virus Indentification Provlems:, Proceedings of the 7th international Virus, Bulletin Conference, P.175-196,1997

[7] http://recaptcha.net/captcha.html

[8] Dax networks, :Teardrop Attack Detection,"Daxnetworks, 2003

[9] John Aycock, Computer Viruses and Malware, 2006

[10] Peter Szor, "The Art of Computer virus Research and Defense, 2005

저자 소개

김성우 (Sung-Woo Kim)

1998 : San Jose State University University Computer Science (학사)
1998~2000: Compaq USA 연구원
2000~2006: Sun Microsystems, USA 연구원
2007~2010: 한국 썬마이크로시스템즈 부장
2010~현재: 현재 서울대학교 계산 과학(석사 과정재학)
2010~현재: 주식회사 아펙스씨앤에스 대표이사
관심분야 : Cloud Computing, Paas, Cloud Infra, Virtualization, BigData, cloud, platform, mobile device, html5

신재인 (Jae-In Shin)

2009년 2월: 한남대학교 컴퓨터 공학과 학사
2011년 8월: 한남대학교 대학원 컴퓨터공학과 석사
2011년 9월~현재: 스컴씨엔에스 연구원
관심분야 : 소프트웨어 공학, 보안공학, 정보보호 컨선설팅 및 위험분석

방영환 (Young-Hwan Bang)

2002년 2월: 대전대학교 컴퓨터공학과 소프트웨어공학전공(공학석사)
2006년 2월: 한남대학교 컴퓨터공학과 시스템소프트웨어공학전공(공학박사)
2006년 5월: 한국과학기술정보연구원 선임연구원
2008년 3월~2010년2월: 클라우드컴퓨팅포럼 사무국장
2010년 3월~현재: ISO/JTC/SC38 국제표준전문위원
2010년 4월~현재: 클라우드데이터센터포럼 운영위원
2010년 11월~현재: 한국생산기술연구원 선임연구원
관심분야: 슈퍼컴퓨팅, 개방형모바일클라우드시스템, 정보보호위험분석평가