특집 04

클라우드 컴퓨팅을 통한 적극적 단말보호방법

목 차

김정훈 · 황용석 · 김성현 · 조시행
(안철수연구소)

1. 서 론

클라우드 컴퓨팅은 로컬 PC나 스마트폰과 같은 단말의 소프트웨어나 스토리지를 사용하지 않고 인터넷을 통해 광범위하게 공유된 자원을 활용하는 컴퓨팅 방법이다. 1960년대 그 개념이 태동되었고 2005년 미국 인터넷 회사인 아마존이 가상화 기반의 클라우드 서비스를 시작하면서 클라우드 컴퓨팅이 본격적으로 사업화되기 시작하였다[1]. 클라우드에서의 보안은 크게 두 가지로 나눠 볼 수 있다. 하나는 서버기반의 클라우드 컴퓨팅 환경에서 발생하는 위협에 대처하는 것이고, 다른 하나는 클라우드 컴퓨팅을 이용한 보안방법에 대한 것이다. 주로 가상화를 이용하여 구성되는 서버기반의 클라우드 컴퓨팅 환경에서는 가상머신을 구성하는 가상디스크파일의 유통과 관리와 가상머신을 호스팅하는 과정 등에서 과거와는 다른 형태의 위협이 존재한다[2]. 이와 같은 새로운 위협에 어떻게 대처할 것인가에 대해 더 많은 논의가 필요하다. 본 논문에서는 후자인 최근 새롭게 떠오르고 있는 클라우드 컴퓨팅 환경을 활용한 단말보호방법(이하 클라우드 시큐리티)을 살펴보기로 한다.

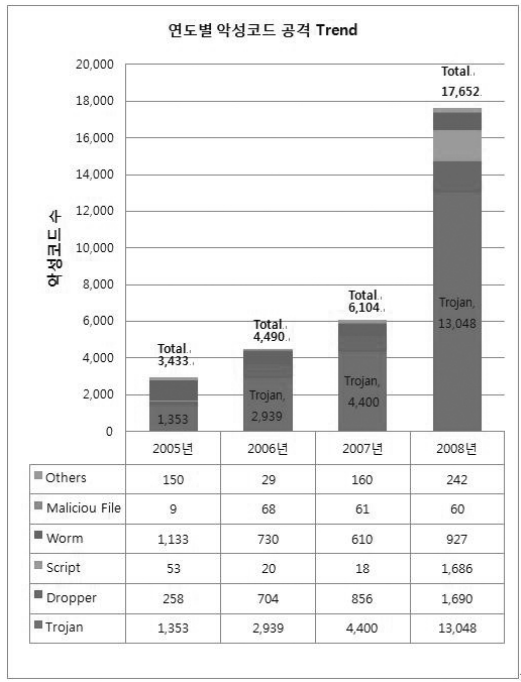
클라우드 시큐리티를 쉽게 표현하면 "벌떼 보안"이 적당할 것이다. 서로 다른 지역에서 서로 다른 상황에 처한 서로 다른 단말이 각각 수집하고 자신의 컴퓨팅 파워를 사용하여 위협을 분석하고 정보 생성한다. 이렇게 생성된 정보를 중앙에 집중하고 많은 컴퓨팅 파워를 사용하여 데이터를 "자동 분석"하여 위협에 조기 대응한다는 것이 기본 개념이다. 각 단말에서 분석과 정보생성이 이루어지고, 생성된 정보는 서버 집단으로 취합되고 재분석된다. 클라우드에 참여한 모든 단말이 위협정보의 수집과 분석에 참여하고 가공된 정보를 공유하게 된다. 마치 꿀벌이 꽃에서 꿀을 따서 벌집에 모으는 것과 같다고 할 수 있다. 여왕벌은 가득 모인 꿀을 영양분으로 알을 낳을 것이고, 알에서 태어난 벌은 다시 꽃으로 날아가 과실을 맺을 수 있도록 꽃을 수정시키고 꿀을 모을 것이다. 시스템에 참여한 모든 단말에서 정보 수집과 분석이 이뤄지고 중앙의 서버와 쌍방향 커뮤니케이션이 이뤄진다. 이를 통하여 각 단말에 발생하는 모든 위협에 대하여 실시간으로 사용자 개입 없이도 대응할 수 있게 된다.

그렇다면 왜 이런 개념이 나오게 되었을까? 사실은 "나오게 되었다"라는 표현보다는 "이제는 이렇게 할 수 밖에 없다"라는 것이 더 정확할 것이다. 과거와 달리 대응속도가 가장 중요한 시대이다[3]. 먼저 현재의 위협 수준에 대해서 알아보자.

2. 현재 악성코드의 위협 수준

단말보호에 있어 첫 번째 위협은 폭발적으로 증가하는 악성코드의 수다. 분석가가 일일이 눈으로 확인하며 악성코드를 분석하던 시대는 이미 몇 년 전에 끝났다. 지금은 하루에도 수없이 많은 의심샘플이 발견되기 때문에 사람이 일일이 확인할 수가 없다. 보안회사가 악성코드를 자동으로 분석하는 것처럼, 공격자들도 자동으로 취약점을 찾고 자동으로 변종을 생성해 낸다. 어렸을 때 "학생과학"과 같은 과학 잡지에서 보던, 로봇이 인간을 대신해서 싸우는 것과 비슷한 느낌을 받는다.

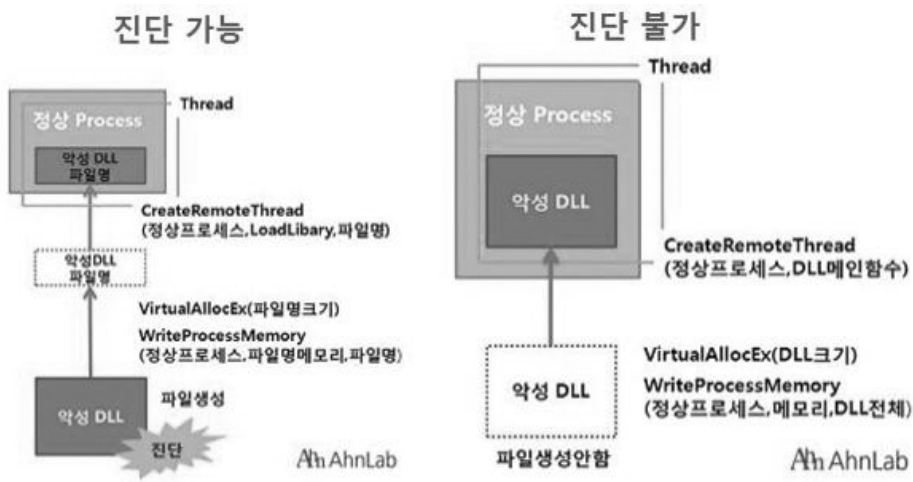
두 번째는 루트킷으로 통칭하는 숨어있는 악성코드다. 과거에 PC나 데이터를 파괴하여 자신의 존재를 과시하는 유형의 악성코드가 점차 사라지고, 정보 침탈을 목적으로 하는 악성코드가 늘어



	2005년	2006년	2007년	2008년
Others	150	29	160	242
Maliciou File	9	68	61	60
Worm	1,133	730	610	927
Script	53	20	18	1,686
Dropper	258	704	856	1,690
Trojan	1,353	2,939	4,400	13,048

(그림 1) 폭발적으로 증가하는 악성코드[4]

나면서 악성코드가 숨기 시작했다. 사용자가 눈치챌 수 없는 것은 물론이고 새로운 기법을 사용한 루트킷은 안티바이러스에도 탐지되지 않는다[5]. 때문에 사용자가 직접 의심 샘플 신고와 같은 어떤 대처를 할 수 있을 가능성이 매우 낮다.



(그림 2) 전통적인 백신으로 진단할 수 없는 메모리 기반 악성코드

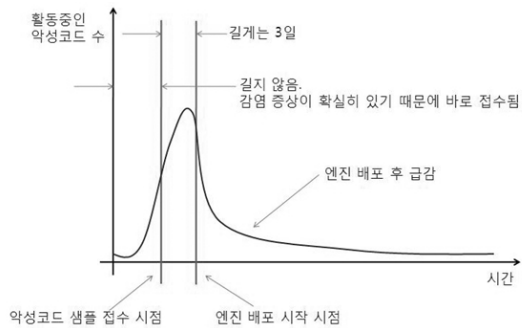
세 번째는 취약점을 이용하여 배포되는 악성코드다. 과거에는 메일에 첨부된 파일이나 인터넷에서 다운받은 또는 메신저로 전달 받은 파일을 직접 실행해야만 악성코드에 감염되는 경우가 많았다. 그러나, 최근에는 "제로데이공격"으로 취약점을 이용한 배포가 줄어들고 있지 않기 때문에 해당 취약점이 고쳐지기 전까지 해당 악성코드의 변종이 무차별 살포되기도 한다[6]. 마치 암살자처럼 소리 없이 살금살금 다가와서 숨어버린다.

현재의 악성코드는 앞에서 언급한 기술들을 모두 결합하여 사용하고 있다. 취약점으로 통해서 침입하고, 침입하면 바로 숨어들고, 자동으로 변종을 생산하여 대량 배포한다. 일단 시스템에 침입하면 스팸메일을 발송하거나, 좀비PC화하여 봇넷(Botnet)을 구성한다. 우리를 더욱 곤란하게 하는 것은 여기에 사회공학적 기법들이 더해지는 것과 소규모의 특정 집단을 대상으로 하는 타겟 공격(Targeted Attack)이다. 사회공학적 기법이 동원되면 자기도 모르게 악성코드를 설치하게 할 뿐만 아니라, PC에 이상행위가 발생해도 정상적인 것으로 인식할 가능성이 있다.

공격당하는 자의 수가 적기 때문에 피해를 입기 전에 공격을 받고 있다고 신고할 가능성이 적다. 특정 정부기관이나 기업을 대상으로 한정하여 공격이 이뤄지는 경우 악성코드를 발견하는데 매우 오랜 시간이 걸릴 수 있다. 게다가 공격대상을 어떤 특정 개인으로 한정하고 사회공학적 기법과 결합한 공격이 발생할 가능성도 있다.

3. 전통적인 악성코드 대응 방법

전통적이고 그리고 현재에도 가장 확실한 또는 가장 정확한 대응 방법은 수집된 샘플을 분석한 후 시그니처를 생성하여 사용자 PC의 악성코드를 진단하는 방법이다. 이는 확실하게 악성으로 확인 된 것만 진단하기 때문에 위협에 대응하는 가장 정확한 방법이다. 이 방법의 단점은 사건이 발생한 후에야 분석이 이루어지고 또 일정 시간이 걸리며, 시그니처를 배포하는데도 시간이 걸린다는 것이다. 따라서, 전통적인 형태의 백신 프로그램은 악성코드가 널리 퍼져 피해가 커지기 전에 진압하는데 그 목적이 있다고 할 수 있다. 다음 그림은 백신 배포에 따른 악성코드 수의 변화를 나타낸다. 악성코드가 확산되기 전에 백신을 배포하여 악성코드에 대응할 수 있다.

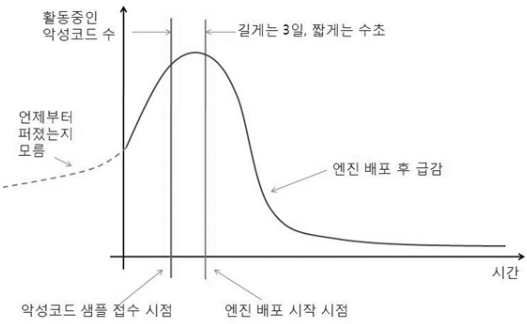


(그림 3) 백신 배포에 따라 변화하는 활동하는 악성코드의 수

과거에는 이렇게 해도 좋았었다. 아니 아주 효율적인 그리고 효과적인 대응방법이었다. 악성코드의 주 목적이 사용자 PC를 파괴하는 것이 주목적이던 시절, 그리고 인터넷이 없었거나 있다 해도 해킹기술과 악성코드가 접목되지 않아 퍼지는 속도가 몇 달씩 걸리던 시절에는 아주 좋은 방법이었다. 악성코드에 감염되면 PC가 망가지거나 느려지는 등의 확실한 증상이 있어서 즉시 신고되었고, 분석가에 의해서 길어도 3일 정도면 분석되어 시그니처로 엔진에 탑재되어 악성코드가 확산되기 전에 진단이 가능했다. 게다가 기술이 점점 발달되어 유사한 악성코드를 자동으로 잡아내는 휴리스틱 진단이나 행동기반탐지 방법 등이 적용되었고, 분석 또한 사람이 직접 하는 것이 아니라 많은 부분을 기계가 자동으로 처리하여 분석시간과 엔진 배포 시간을 획기적으로 줄여 나가기 시작했다. 현재 탑 클래스의 벤더는 접수된 의심샘플을 분석하고 엔진

으로 배포하기까지 수 십분 이내에 가능하다고 한다[7].

그러나 지금은 상황이 달라졌다. 샘플이 수집되면 수 십분 내에 엔진이 배포되지만 샘플이 수집되기까지 오랜 시간이 걸린다. 악성코드에 감염되어도 증상이 나타나지 않는 경우가 많기 때문이다. 제아무리 빠른 자동 분석 시스템을 갖추고 있다 해도 샘플이 접수되지 않는데 무슨 소용이란 말인가? 악성코드에 대한 대응 시간은 수집에 걸린 시간 + 분석에 걸린 시간(수 초에서 수 일까지) + 엔진 배포 시간(수 분에서 수 시간)이다. 이 중에서 분석에 걸린 시간과 엔진 배포에 걸린 시간은 넉넉히 잡아서 각각 3일과 한 시간 정도로 한정 시 킬 수 있는 값이지만, 수집에 걸린 시간은 얼마가 될지 알 수가 없다. 앞에서 언급한 것처럼 샘플을 보고할 사용자가 감염되었는지를 알기 어렵기 때문이다. 따라서, (그림 3)의 시간에 따른 악성코드의 대응은 다음과 같이 다시 그려져야 한다.



(그림 4) 다시 그려진 시간에 따른 활동하는 악성코드의 수

지금 이 순간 우리 PC와 우리의 넷북, 앞으로는 스마트폰에도 우리가 모르는 악성코드가 몰래 숨어서 동작하고 있을 수 있다. 실제로 현실에서 일어나고 있는 일이다. 사회적으로 크게 이슈화 되진 않았지만 작년에 실제로 있었던 사례를 들어보고자 한다. 요약해서 설명하면 이렇다. 어떤 사용자로부터 분석 요청이 들어왔다. PC가 뭔가 좀 이상하다는 것이다. 대응팀에서 원격지원을 했으나 별 다른 악성코드를 발견하지 못했다. 하지만 약간의 이상 증상은 계속 나타났다. 더욱 자세한 분석을 위해 하드디스크의 사본을 제작하는 과정에서 우연히 작고 이상한 코드가 발견되었고 오랜 시간 분석을 거친 결과, 기술적으로 고도로 발전된 악성코드란 것을 확인했다. 이 악성코드가 바로 Win-Trojan/TdlRootkit[8]이다. 문제는 이 악성코드가 얼마나 오래 전부터 PC에 있었는지 아무도 모른다는 것이다. 문제의 악성코드와 같은 이름의 이상한 파일에 대한 떠도는 소문이 있었던 걸로 유추해 보아서 막연히 꽤 오랜 시간 잠복한 했을 거라 추정할 뿐이다.

이제는 과거와는 다른 형태의 보다 적극적이고 능동적인 대응방법을 사용해야 할 때다. 악성코드가 컴퓨터에 손상을 주는 형태보다는 악성행위를 하지 않고 조용히 어떤 정보를 탈취하거나 빠르게 치고 빠지는 1회용 악성코드(one-time malware) 형태로 변화하고 있기 때문에 과거와 달리 정보 수집 범위와 대응 속도가 매우 중요해 졌다. 악성코드보다 더 빨리 움직여야 한다. 그리고 악성코드가 몰래 숨어들어오는 만큼 작은 의심행위에도 주의를 기울여야 한다. 위협을 경고하고 사용자의 허가를 요청하는 방식의 HIPS(Host-based Intrusion Prevention System)가 하나의 대안이 될 수 있으나 누구나 사용할 수 있는 형태 그리고 전체 인터넷 망에서 발생하는 위협에 대해 상호 협조하여 신속히 대응할 수 있는 새로운 시스템이 필요하다.

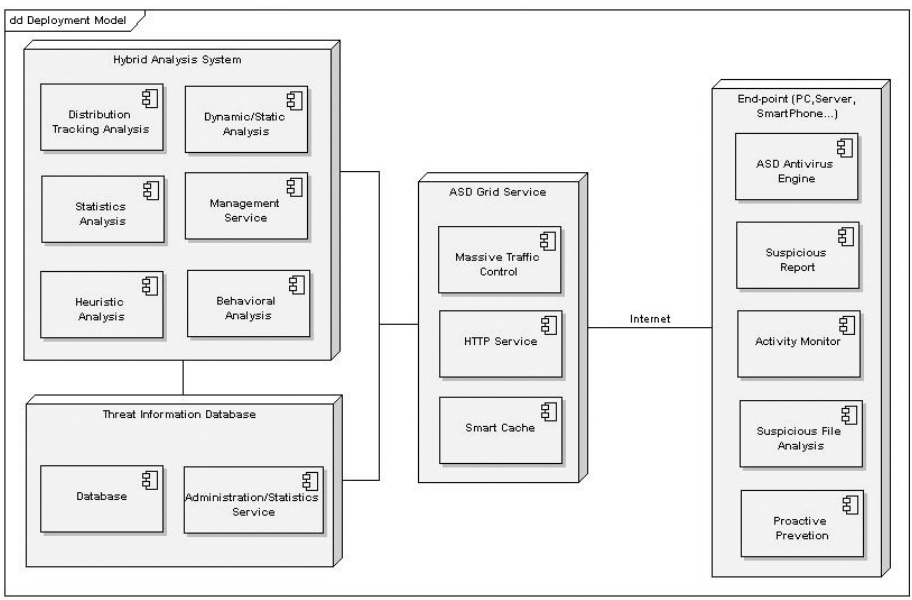
4. 단말보호를 위한 클라우드 컴퓨팅 적용사례

클라우드 시큐리티는 단방향으로 정보를 전달하는 형태의 전통적인 안티바이러스 제품과는 달리 양방향으로 정보를 주고 받는다. PC나 스마트폰과 같은 어떤 단말에서 의심행위를 탐지하고 중앙으로 보고하면 중앙의 종합위협분석시스템이 보고된 어떤 정보를 통계적 방법을 포함

한 다양한 방법으로 실시간 분석 재가공하고 다시 PC나 어떤 네트워크 보안 장비와 같은 각 단말로 내보낸다. 양방향으로 적극적으로 정보를 교환하는 것이다. 클라우드 시큐리티 네트워크에 참여한 하나의 단말에서 보고한 위협 정보는 재가공되어 전체 시스템이 그 혜택을 보게 된다. 또 그 단말도 다른 단말이 보고한 위협 정보의 혜택을 공유한다. 클라우드 시큐리티의 초기 형태는 내 PC에서 발생한 의심스런 행위나 데이터 그리고 분석되지 않은 파일을 분석서버에 보내서 실시간으로 분석결과를 받는 것이다. 분석결과에 따라 위협으로 판단하고 격리할 것인지 허용할 것인지가 정해진다.

안철수연구소는 2009년 ASD(AhnLab Smart Defense)라는 클라우드 시큐리티 기술을 선보였다. 클라우드 시큐리티를 구현함에 있어서 세 가지 핵심 기술이 필요하다. 위협정보에 대한 자동 분석 기술과 대규모 사용자를 처리할 수 있는 서버기술 그리고 단말에서 의심행위를 탐지하고 대응할 수 있는 안티바이러스 기술이다. ASD는 전통적인 백신의 기능과 행위 탐지 기능, 사전 방역 기능과 같은 단말 보안의 기술과 서버 쪽의 대용량 트래픽 처리 기술과 대용량 분석기술, 데이터베이스 관리 기술과 같은 서버 쪽 기술이 결합되어 구성된다. 서버 쪽 아키텍처는 위협 분석을 담당하는 HAS(Hybrid Analysis System)와 클라이언트와의 통신을 담당하는 Grid Service, 위협 정보 데이터베이스로 구성되어 있다. HAS는 위협자동분석 시스템으로써 각 단말이 보고한 위협정보에 대하여 정/동적 분석과 행위분석, 통계적 분석, 휴리스틱 분석 등을 실시간으로 자동 분석한다. 이렇게 분석된 정보는 위협 정보 데이터베이스에 실시간 반영된다. 위협 정보 데이터베이스는 HAS에서 분석된 각종 정보를 실시간으로 반영하고, 행위정보와 파일정보에 대하여 WhiteList와 BlackList로 구분한다. 가공된 정보는 Grid Service를 통해 단말의 안티바이러스 엔진에 제공되게 된다.

오늘날 고속 인터넷 망의 보급과 하드웨어 가격의 하락은 온라인으로 분석을 요청하고 결과

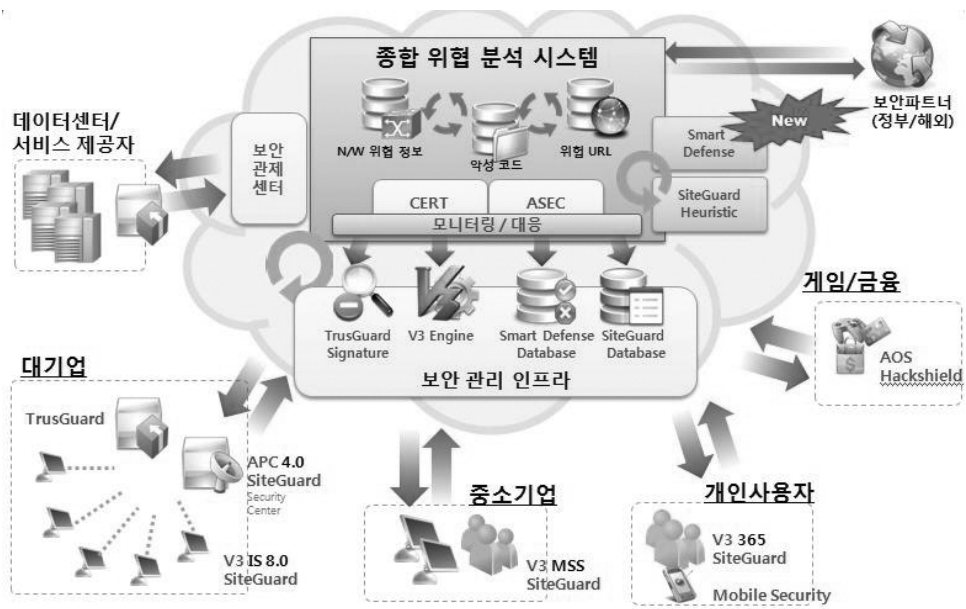


(그림 5) ASD(AhnLab Smart Defense) Technology Architecture

를 받는 거대한 네트워크 시스템의 구축을 가능케 하였다. 많은 수의 장비를 동원한 막강한 컴퓨팅 파워로 대부분의 악성코드와 그 변종은 수초에서 수 시간 내에 분석된다. 서버에서는 보고된 파일에 대하여 반복적으로 분석을 시도하기 때문에 새로운 방식의 악성코드가 발견되었을 때, 사용자PC에서 별다른 액션을 취하고 있지 않아도 과거의 유사한 집단 또한 즉시 분석되어 사용자 PC에서 바로 진단된다. 새로운 악성코드나 취약점에 대한 정보가 분석서버에 지속적으로 업데이트 되기 때문에 가능한 일이다. 이와 같이 적극적이고 능동적으로 대응하는 것만이 오늘날 기술적으로 고도화되고 사회공학적으로 사용자를 기만하는 악성코드를 방어하는 유일한 방법이다. 이러한 형태의 보안은 비단 안철수연구소만 하는 것은 아니다. 동일하진 않지만 시만텍, 카스퍼스키와 같은 탑 클래스 보안 업체뿐만 아니라 MS도 SpyNet이란 이름으로 시큐리티 네트워크 또는 시큐리티 유저 서포터즈와 같은 형태를 운영하고 있다[9,10].

5. 결 론

최초의 PC 바이러스 Brain이 발견되지 20년이 넘는 동안 발견된 악성코드에 대해 후속 대응하는 것에 주력해왔다. 오늘날의 보안 위협 즉 다양한 변종과 취약점을 이용한 제로데이(zero-day) 공격, 금전 침탈을 위한 타겟팅 된 공격 등은 이미 수 년 전부터 여러 연구자들에 의해서 예견되어 왔다. 그리고 지금 이 순간 실제의 위협으로 다가오고 있다. 이를 대비하여 휴리스틱 기법이나 행위기반의 사전 방어 기능, HIPS (Host-based Intrusion Prevention System) 등이 나왔으나, 이들 각각만으로는 오늘날의 위협에 대해서 효과적으로 대응할 수 없었다. 클라우드 컴퓨팅은 기존의 대응 패러다임을 사후-대응에서 사전-대응으로 완전히 바꿔줄 수 있는 기회를 제공한다. 발견되는 악성코드에 대응하기에 급급했던 지난날과 달리 악성코드가 나타나도 양방향으로 정보를 교환하고 분석하여 즉시 꼼짝 못하게 하는 기술이 바로 클라우드 시큐리



(그림 6) 안랩의 ACCESS 전략(AhnLab Cloud Computing E-Security Strategy)

티다. 서로 단절된 개별 단말에서 운용되던 기술을 클라우드 컴퓨팅으로 하나로 묶는 것에 대해서 더 많은 연구가 필요하다. 클라우드는 개별 기술을 결합하여 시너지를 낼 수 있도록 하는 플랫폼이다. 다양한 연구가 결실을 맺어서 악성코드로부터 해방되는 그 날을 꿈꾸어 본다.

참고문헌

[1] wikipedia, Cloud Computing, http://en.wikipedia.org/wiki/Cloud_computing

[2] Joanna Rutkowska, Virtualization - the other side of the coin, 2007

[3] 안철수연구소, 악성코드 대응 기술의 새로운 패러다임 AhnLab Smart Defense, 2009

[4] 안철수연구소, ASEC Report

[5] 김지훈, 날찾지마!탐지/진단을 피하기 위한 악성코드의 발전, http://www.ahnlab.com/kr/site/securityinfo/secunews/secuNewsView.do?curPage=1&menu_dist=3&seq=15722&columnist=0&dir_group_dist=0&dir_code=, 2010

[6] 김지훈, 틈새로 공격한다. 제로데이 취약점 공격, http://www.ahnlab.com/kr/site/securityinfo/secunews/secuNewsView.do?menu_dist=3&seq=15748, 2010

[7] Symantec, The Digital Immune System

[8] Nguyễn Phố Sơn, TDL3: Part I Why so serious? Let's put a smile … A detailed analysis of TDL rootkit 3rd generation

[9] http://windows.microsoft.com/en-US/windows-vista/Join-the-Microsoft-SpyNet-community

[10] http://www.symantec.com/about/profile/policies/ncwprivacy.jsp

[11] 안철수연구소 클라우드 보안 서비스 전략, http://www.ahnlab.com/kr/site/etc/popSecurityMap.do

저자약력

김 정 훈

2003년~현재 안철수연구소 기반기술팀 수석연구원
관심분야 : 클라우드 컴퓨팅, 네트워크 보안, 정보보호
이 메 일 : kimjh@ahnlab.com

황 용 석

1998년 2월 건국대학교 항공우주공학과(학사)
2002년 2월 건국대학교 항공우주공학과(석사)
2004년~현재 안철수연구소 기반기술팀 선임연구원
관심분야 : 정보보호, 단말보안, 클라우드 컴퓨팅
이 메 일 : hwang@ahnlab.com

김 성 연

1996년 2월 국민대학교 전자공학과(학사)
1998년 2월 국민대학교 전자공학과(석사)
1999년~현재 안철수연구소 기반기술팀 팀장
관심분야 : 클라우드 컴퓨팅, 루트킷, 정보보안
이 메 일 : shkim@ahnlab.com

조 시 행

1984년 2월 한양대학교 건축공학과(학사)
1986년~1991년 (주)쌍용컴퓨터 시스템연구소
1992년~1995년 한컴퓨터주식회사
1996년~현재 안철수연구소 상무
관심분야 : 정보보호, 보안
이 메 일 : shcho@ahnlab.com