중소기업융합학회 논문지 제4권 제2호 pp. 13-18, 2014 ISSN 2234-4438

악성코드 분석 및 대응 방안

홍성혁[1*]
[1]백석대학교 정보통신학부

Analysis and Countermeasure of Malicious Code

Sunghuyck Hong[1*]
[1]Division of Information and Communication, Baekseok University

요 약 본 각종 PC 및 스마트폰의 발전으로 인하여 인터넷이 발전함으로써 정보시스템과 인터넷과 스마트폰의 발전으로 인하여 모든 정보 자산의 네트워킹이 가능해졌다. 이로 인해 각종 취약점과 프로그램을 악용하여 악성코드를 만들어 정보를 빼내가거나, 유출을 시키는 등의 범죄가 하루가 다르게 늘어나고 있다. 이에 따른 각 시스템별 보안위협 및 어떤 방법으로, 어떤 경로로 위협을 해오는지를 알아보고 이에 따라 시스템을 비교하여 목적 및 수단을 알아보고 취약점을 분석하고 대응책을 알아본다.

Abstract Due to the development of information systems and the Internet, the Internet and smart phones can access networking in any where and any time. This causes the program to exploit various vulnerabilities and malicious code created to go out information, the disclosure of such crime increasing day by day. The proposed countermeasure model will be able to contribute to block all kinds of malicious code activities.

Key Words : Malicious code, virus attack, network security, network monitoring

1. 서론

현재 각종 PC 및 스마트폰기기의 발전으로 인하여 인터넷이 발전함으로써 이를 악용 하는 사람들로 인하여 각종 악성코드가 개발이 되어 각종 기기들을 위협하고 있다. 본 논문에서는 각종PC와 스마트폰기기의 발전으로 인하여 확산되고 있는 소셜 네트워크 서비스, 가상화 기술 등의 빠른 확산으로 급격하게 늘어난 새로운 형태의 악성코드를 분석하고 악성코드에 의해 발생된 보안위협 사례를 분석하여 본 논문에서 각종 악성코드에 대하여 알아보고 이를 예방하는 방안에 대하여 살펴본다.

2. 보안위협 변화

2.1 보안 위협

현재 PC와 스마트폰의 사용 확산으로 인하여 인터넷 사용량이 급격히 증가하고 있다. 태블릿PC와 스마트폰을 업무로 사용하는 직원의 수가 2014년 전 세계 직장인들을 대상으로 조사에 따르면 추후에는 기기를 업무용으로만 사용하는 인원이 3억 5천만 명에 달할 것으로 전망되고 있으며[1]. 2015년에는 4,200만 명에 달할 것으로 전망하고 있다. 이에 따른 각종 네트워크를 위협하는 보안위협사례가 크게 증가하여 안드로이드에서 2012년 4/3분기에 51,477개의 악성코드가 발견되었다. 이 수치는 직전 분기 대비의 약 10배가 증가된 수치이다. 현재 이미 많은 기업들이 BYOD 허용 여부를 떠나서 이미 많은 직원들이 스마트폰 및 태블릿PC를 많이 활용함에 따른 보안위협 도 크게 증가하고 있다. 이에 따른 취약점 분석 평가, 보안위협에 대한 고려 및 대응책이 필수적으로 요구되어지고 있다.

Received 2014-3-2 Revised 2014-5-26 Accepted 2014-5-26
*교신저자 : Sunghuyck Hong(sunghyuck.hong@gmail.com)

2.2 제어시스템을 노리는 보안위협의 증가

최초로 제어시스템을 공격하였던 건 2011년 10월 발견되었던 Stuxnet이다. 이 Stuxnet은 최초의 사이버 무기로 평가가 되고 있다.[2]. 그 이후에도 기반시설 및 제어시스템을 공격 대상으로 하는 대표적인 악성코드들이 계속 발견이 되어 지고 있다. 현재 미국은 ICS-CERT(Industrial Control System - CERT)[3]를 운영하고 있다. 이 ICS-CERT Monthly Report에 따른 통계로는 매년 제어시스템 사이버 사례의 증가는 아래 <Table 1> ICS Incident & Vulnerabilities를 보면 알 수 있다.

Table 1. ICS Incident & Vulnerabilities

ICS-CERT Metrics	FY-10	FY-11	FY-12
ICS Incident	40	130	198
ICS Related Vulnerabilities	17	145	171

FY: Fiscal Year = 회계연도

위 <Table 1>과 같이 제어시스템 및 기반시설에 대한 보안위협은 계속 되어 지고 있으며, 통계자료를 분석함과 같이 앞으로도 보안위협은 더욱더 증가 할 것으로 예상한다. 이에 대한 대응책은 체크리스트를 만들어 주기적으로 제어시스템에 취약점을 평가 및 분석을 하여 체크를 하며 이에 따른 보안대책을 수립하여 보안사례가 발생하지 않도록 노력을 해야 한다.

2.3 소셜 네트워킹 환경을 노리는 보안위협

현재 스마트폰의 엄청난 확장을 이루게 해준 최고의 아이템은 소셜 네트워크 서비스 (Social Network Service, 이하 SNS)이다. SNS는 전 세계의 다양한 사람들이 서로 다양성을 존중하며 개인과 기업 간의 소통의 장으로 자리매김을 하고 있다. 전 세계적으로 대표적인 SNS서비스를 보면 '트위터'와 '페이스 북'을 예로 들 수 있다. 이러한 SNS는 전 세계적으로 소통을 할 수 있는 플랫폼으로 발전이 되어 가고 있다.

하지만 이러한 SNS의 사용이 많이 짐과 동시에 보안위협 또한 많아지고 있다. 이에 따른 SNS 환경에서의 보안사례 및 역기능들을 살펴보자

2.3.1 SNS의 보안사례 및 문제점

SNS는 전 세계의 사람들이 사용을 하기 때문에 이메일, 메신저 등을 이용하여 보다 강력하게 사람들을 현혹시킬 수 있는 성인광고, 성인물들 같은 광고를 위한 스팸봇과 현재 골칫덩이로 전락한 트로이목마와 같은 정보유출용 악성코드 유포, 개인정보 및 카드정보를 유출하기 위한 피싱 웹사이트로 링크를 전송하는 등의 많은 피해를 발생 시키고 있다.

아래 <Table 2>는 SNS상의 보안위협 사례들을 나타낸다.

Table 2. SNS상의 보안위협 사례

목적	수단 (내용)	년도/월
악성코드유포	트위터 초대메일 위장	09/06
서비스거부	트위터 서버스 공격	09/08
봇넷C&C	트위터 계정 활용	09/08
봇넷C&C	구글 뉴스그룹 활용	09/09
악성코드유포	페이스북 암호변경 요청	09/09
악성코드유포	트위터 초대메일 위장	09/09
악성코드유포	페이스북 암호변경 요청	09/10
악성코드유포	페이스북 피싱 웹페이지	10/02
악성코드유포	트위터 다이렉트 메시지로 피싱 링크 전송	10/02
스팸메일	트위터 다이렉트 메시지	10/02
신용카드정보	트위터 메시지를 이용한 메신저 추가 요청	10/03
악성코드유포	페이스북 관리자로 위장	10/04
스팸메일	트위터 메일로 위장한 성인약품 광고	10/05
허위백신유포	구글 그룹스 활용	10/05
봇넷C&C	트위터 계정 활용	10/05
악성코드유포	단축URL 활용	10/05
허위백신유포	트위터 암호변경	10/06
악성코드유포	페이스북 위장	10/06

또, 작년부터 지속적으로 C&C(악의적인 봇넷의 명령, 제어 서버)가 활용됨에 따라서 SNS 환경에는 꾸준한 보안위협에 대한 새로운 시도가 멈추지 않고 계속 만들어지고 있음을 확인해 볼 수 있다. 인터넷을 사용하는 사람이라면 PC와 모바일을 막론하고 누구든지 트위터나 페이스북과 같은 소셜 네트워크를 통해 자신의 일상이나, 자신의 취미, 사진 등을 올리게 됨으로써 정보가 오픈이 된 상태에서 서로를 신뢰함으로써 소통을 하기 때문에 자신도 모르게 개인 정보들이 어디론가 빠져 나가게 될 수 있다. 그러므로 공격자들은 SNS를 통하여 수많은 공격대상을 만들 수 있고 공격대상으로부터 정보를 빼낼 수 있다. 이에 따른 대응방안은 모바일이나 PC사용 중 메시지에 포함이 되어 있는 URL 접근을 주의하고, 지인의 메시지라도 다시 한 번 확인을 해보고 열람을 하여야

하고 검증되지 않은 서비스의 이용은 가급적이면 사용하지 않도록 해야 한다.

3. 악성코드 종류

악성코드를 분류하여 악성코드의 다양한 종류와 악성코드의 특성을 확인하여 분석해보고 어떤 목적으로 사용되고, 어떤 방법으로 사용되는지 알아 볼 수 있다.

3.1 대표적 악성코드 분류

3.1.1 Trojan Horse

Trojan Horse란 악성 루틴을 포함하고 있으면서 정상적인 하나의 프로그램으로 위장을 하고 있는 프로그램이다. 다른 사용자가 알 수가 없도록 포함되어 있으며 자기 스스로는 복제를 할 수가 없다. Trojan Horse는 Worm이나 Virus와는 다르게 공격자가 악의적 목적을 가지고 삽입 시킨다. 주로 DDos 공격, Backdoor설치를 통하여 개인 정보를 빼내는 행위를 한다. Trojan Horse는 자기복제가 불가능해서 다른 프로그램에 자기 자신을 집어넣어 사용자가 그 프로그램을 실행하게 만들어 피해를 유발시키는 악성코드이다.

3.1.2 Worm

Worm이란 자기를 복제할 수 있는 능력을 가지고 있고, 혼자서 컴퓨터와 컴퓨터를 이동해가며 전파시키거나, 프로그램과 프로그램 사이를 이동하며 전파를 시킨다. Worm은 자신을 복제하거나, 사용자가 인지할 수 없는 방법으로 이메일을 전송한다던가, 프로그램이 배포되기 전 정상적 파일에 새로운 코드들을 넣는 등의 수행을 하는 악성코드이다. Worm은 자기 자신 자체만으로도 네트워크를 타고 전파가 가능하다.

3.1.3 Spy-ware(Spy Software)

Spy-ware는 주로 무료로 배포되는 소프트웨어 프로그램에 포함이 되어 있다. 감염되어있는 네트워크나 컴퓨터 안에서 동의없이 개인이나 기업에 대한 정보를 빼내어 공격자에게 보내도록 제작이 되어있는 프로그램이다. Spy-ware의 주요 악성행위는 각종 정보, 데이터, 금융정보, 개인정보 등을 빼내는 행위를 한다.

3.1.4 Bot

Bot은 공격자가 사용자의 컴퓨터를 제어할 수 있도록 만들어 주는 프로그램으로써 Bot Master의 명령에 따라서 다양한 경로로 사용자의 PC에 침입을 한다. Bot Master는 Bot에 감염된 PC를 자유자제로 조종할 수 있고, PC에 저장되어 있는 정보를 빼낼 수가 있어서 정보유출도 쉽게 할 수 있고, 다른 시스템 들을 공격을 하는 목적으로 사용된다. botner는 여러 Bot들이 모여 하나의 네트워크를 형성한 경우이다.

3.1.5 Backdoor

Backdoor란 시스템에 접근을 할 때 정상적인 인증절차 등을 거치지 않고 시스템이나 응용프로그램으로 접근을 할 수 있도록 만들어주는 도구이다. Backdoor는 일종의 통로이다. 공격자가 시스템으로 최초로 침입을 한 후에, 공격자가 원할 때 마다 쉽게 다시 침입을 하여 권한을 획득 할 수 있다.[5]

4. 악성코드 분석

악성코드를 분석하는 방법으로는 동적 분석(Dynamic Analysis)방법과 정적 분석(Static Analysis)방법으로 나뉠수 있다. 이런 방법으로 악성코드를 분석을 함으로 악성코드를 사용하여 공격하려는 시스템 내 동작 과정, 전파 경로, 및 악성코드가 하고자 하는 목적을 빼내기 전 정보를 판단하여 해당 악성코드에 대하여 대응할 수 있다.

4.1 동적 분석 방법

동적 분석 방법은 실제로 악성코드를 실행시키고 악성코드가 수행될 때 그 내용을 분석하는 방법이다. 가상머신(Virtual Machine)이나 에뮬레이터(Emulator)에서 파일이나 프로세스 및 네트워크를 이용하는 행위들을 위한 API(Application Programming Interface) 호출 등의 분석을 실시간으로 하는 방법이다. 동적 분석 방법은 비교적으로 악성행위 분석이 정확하게 가능하다. 악성코드가 특별한 환경이나 조건을 만족하지 않지만 동작을 하기 위한다는 조건이 만족하지 않을 때에는 분석이 어려울 수가 있다. 단점으로는 여러 실행 경로 중 소수의 경로만 분석을 할 수 있다. 에뮬레이터나 가상 머신과 같은

분석 환경이 구축이 되면 악성코드에 대한 동적 분석을 수행 할 수 있다[6]. 장점으로는 정상적인상태로 감염 전에 빠른 복구를 할 수가 있다. 이 때문에 안전하게 분석을 할 수가 있다. 동적 분석은 주어진 악성코드에 대한 API 후킹을 분석 환경에서 할 수가 있다. 여기서 API후킹은 응용프로그램으로부터 호출이 되어지는 API를 가로채어서 악성코드를 분석하는 사람이 작성이 되어있는 응용프로그램의 함수들을 처리를 할 수 있도록 해주는 메커니즘이고 개발언어에 대해 제약 없이 적용할 수 있기 때문에 역추적이나, 디버깅, 모니터링 등에서 사용 할 수가 있다. Windows 운영체제는 기본적인 3개의 DLL(Dynamic Linking Library), kernel32.dll과 user32.dll과 gdi32.dll에 대부분의 API를 제공하고, 또 구현을 한다. 응용프로그램은 실행할 때 자기 자신의 프로세스 주소 공간으로 DLL을 맵핑을 하고 나서 사용을 하게 된다.[7] 네트워크 연결 및 파일, 프로세스, 레지스트리 등을 추적은 API가 악성코드가 실행이 됨에 따라서 호출이 되는 것을 모니터링을 수행함으로 추적을 할 수가 있다. 아래의 <Table 3>은 악성 코드의 행위에 따른 분석을 할 때 이용 할 수 있는 API들의 예를 보여준다.

Table 3. 악성 행위 분석에 이용이 가능한 API의 예

Target	APIs
File	CreateFile(), CopyFile(), GetModuleFileName()
Registry	RegCreateKeyEx(), RegCreateKey(), RegOpenKeyEx(), RegOpenKey(), RegSetValueEx(), RegSetValue()
Process	CreateProcess(), TerminateProcess(), WinExec()
Network	Send()/recv(), inet_ntoa(), connect(), Gethostbyname()

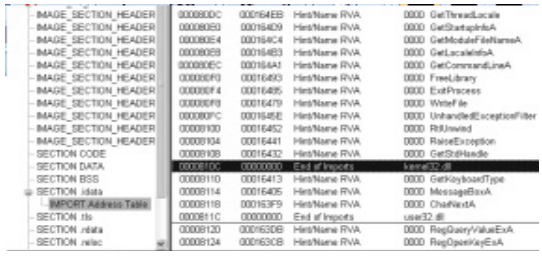
최근에는 악성코드를 자동으로 분석하여 분석 결과를 알려 주는 서비스가 점점 많아지고 있다. 제공 되어지는 악성코드의 분석 내용을 보면 레지스트리, 네트워크, 프로세스 파일 등의 변화에 대한 로그들을 기록이 되어 있다. 그리고 변경이 되는 파일의 경로파악 및 파일명을 알 수 있고, 악성코드가 접속을 시도하려하는 특정한 IP주소, 도메인 등의 네트워크로 접속을 시도할 때 이에 대한 정보를 나타내 준다. 아래의 <Table 4>는 악성코드를 동적 행위 분석 을 할 수 있도록 서비스 해주는 웹사이트들을 나타낸 것 이다.

Table 4. 동적 행위 분석 서비스 웹 사이트

Name	URL
CWSandbox	http://mwanalysis.org/
ANUBIS	http://anubis.iseclab.org/
Zero Wine	http://zerowine.sourceforge.net.
Threat Expert	http://www.threatexpert.com/
Xandora	http://xandora.security.net.my/
Norman Sandbox	http://www.norman.com/
Joebox,	http://www.joebox.org/

4.2 정적 분석 방법

정적 분석 방법은 디컴파일러(Decompiler)나 디스어셈블러(Dis-assembler)를 이용하여 역공학(Reverse Engineering)을 통하여서 악성 바이너리에 대한 어셈블리 명령어를 악성코드 바이너리 파일을 실제로 실행을 시키지 않은 채로 생성하고 분석하는 방법이다. 정적 분석 방법은 안전한 분석 방법이다. 왜냐하면 악성코드를 직접 실행을 시키지 않기 때문이다. 단점으로는 자동화가 어렵고, 시간과 노력이 많이 필요하여 시간이 많이 소모가 된다는 점과, 암호화가 되어 있는 등의 방법으로 패킹(packing)이 되어 있을 경우에는 분석이 어렵다는 점이 있다. 정적 분석을 수행할 때 먼저 분석해야 할 것은 Windows 바이너리 형태로 되어있는 PE(Portable Executable) 파일의 내용 및 구조이다[8]. 각 섹션별로 정보의 요소들을 추출해야 하고, 바이너리로 구성이 된 PE 파일을 분석해야 한다. [Fig. 1][5]과 같이 PE 파일의 IAT(Import Address Table)에 포함되어있는 API 정보를 추출함으로 악성코드가 어떤 행위들을 수행하고 있는지를 파악 및 포함 되어있는 관련 정보에 대하여 추측 가능하기 위해 텍스트 스트링을 추출을 한다.

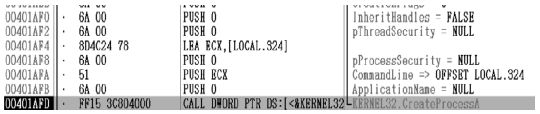


[Fig. 1] Import Address Table의 예제

또한, [Fig. 2]처럼 PE 파일을 어셈블리 명령어로 변환을 하여 코드 블록을 작은 단위로 분할할 수 있고 악성코

`* 1) API 2) 정적 분석 방법

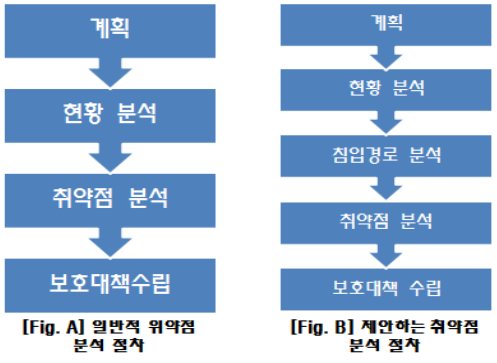
드의 동작 및 흐름을 이 코드 블록의 제어 흐름을 통하여 파악을 할 수가 있다.



[Fig. 2] 디스어셈블의 예제

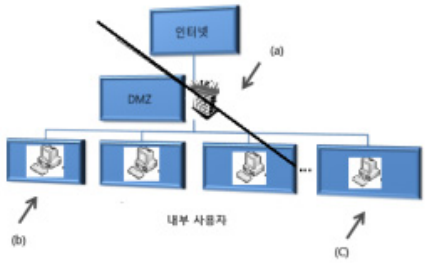
4.3 사이버보안을 위협하는 악성코드의 변화 및 이에 따른 취약점 분석

사이버보안위협의 증가로 인한 취약점을 분석하기 위해서는 보안항목을 구성하는 것 외에 보안에 대한 절차에 대한 구체적은 구조를 개선하는 방법도 필요하다. 일반적으로 취약성 분석은 취약점 분석, 현황 분석 수행, 보호대책의 수립, 계획수립으로 이루어지고 제안하는 취약점 분석은 취약점 분석을 세분화시켜 취약점 분석 방법을 선정, 침입경로 분석, 취약점 분석으로 구성을 한다. 이를 구성하여 사이버를 위협하고 있는 악성코드들의 변화에 대응을 하고 있다.



[Fig. 3] 제안하는 취약점 분석 절차

4.4 침입경로 분석



[Fig. 4] 침입경로의 다양화

일반적인 네트워크의 구조를 간단하게 표현 하여 나타낸 [Fig. 4]를 참조하며 본 장을 분석해보자. 우선 그림 [Fig. 4]을 보고 검은 선 위쪽을 외부 네트워크로 보고, 외부로부터의 침입경로인 (a)만 분석을 한다. 검은선의 아래쪽은 내부 네트워크로 인식하고 USB 등 외부 기기를 통한 악성코드 감염 등에 대해서만 고려하여 분석을 한다. 이에 따라 외부에서부터 보안대책, 접근통제는 방화벽 과 같은 유선네트워크 접점에서만 적용이 된다. 그러나 BYOD,나 모바일 인터넷의 사용 증가로 인하여 전통적인 내부와 외부의 구분을 의미가 없도록 만들고 있다. 내부 네트워크로 인식이 되던 사용자 단말기에서 직접적으로 모바일의 인터넷을 연결 할 수 있도록 하여 외부로부터 내부로 침입을 할 수 있으며, 방화벽을 무력화 시키는 방식의 테더링(tethering)이 있다. 이에 따른 사이버 보안 위협에 대한 변화에 따라 일반적 취약점 분석시, 외부네트워크를 구분하고 연결 접점을 분석을 하는 의미가 없어졌다. 대신 [Fig. 4]의 (b)와 ©와 같이 내부 사용자의 영역에서도 침입경로가 외부로부터 형성될 수 있다는 것을 인식하고, 다양한 침입경로에 대하여 분석도 필요하다. 또 보안대책을 수립할 때에도 기존의 내부, 외부 구분에 따른 입체적이고 전방위적인 보안대책이 필요 하다.

5. 향후 연구 방향

앞으로는 사이버공간을 위협하는 악성코드들이 더욱 더 많이 만들어 질 것으로 본다. 본 논문을 작성 하면서 느끼게 된 것은 프로그램이 개발이 됨에 따라 악성코드는 제작이 될 것이고, 이 악성코드에 대한 백신을 만든다 해도 또 다른 악성코드로 인하여 사이버 공간은 위협을 받게 될 것이다. 이에 따른 대응을 생각을 해보면 최근 유행하고 있는 스미싱과 같은 악성코드가 담긴 URL을 포함한 메시지나, 메일, 등을 받으면 검증되지 않은 것이면 절대로 사용하지 않도록 하고, 주변 지인의 이름으로 온 문자라 하더라도 다시 한 번 문자를 보낸 사람에게 확인의 절차를 거치고 이용해야 한다. 그리고 프로그램을 사용 할 때에는 번거롭더라도 운영 가이드라인을 확인해 보고, USB와 같은 외부기기들을 연결했을 때 들어 올 수 있는 악성코드를 대비하여 백신프로그램으로 점검 후 사용을 할 수 있도록 한다. 그리고 사이버공간이 활성화가 됨으로써 보안을 대응 할 수 있는 조직이 필요하다는 생각만을 가지고 있다가는 악성코드로부터 승리를 할 수가

없을 것이다. 악성코드가 발전하는 것처럼 우리도 이에 대응을 했던 경험을 토대로 더욱 더 발전을 해나가야 한다. 앞으로는 멀지 않은 미래에 대해서도 언제나 공격자보다 더욱더 앞선 방어 시스템을 구축하여 앞으로의 공격에도 절대지지 않길 바라며 깨끗한 사이버공간을 계속 유지해 나가길 희망해본다.

참고문헌

[1] “Mobile Security: BYOD, mCommerce, Consumer & Enterprise 2013-2018”, Juniper research, 2013.
[2] Nicolas Falliere, Liam O Murchu and Eric Chien, “W32.Stuxnet Dossier”, Symantec Security Response Report, Feb. 2011.
[3] http://ics-cert.us-cert.gov
[4] DOI: http://dx.doi.org/10.1016/j.ijrefrig.2006.03.005
[5] 강 부중, 한 경수, 임 을규, 악성코드 현황 및 탐지 기술, Han Yang Univ. 2012.1, p.44-53
[6] 서정택, 가상환경을 이용한 악성코드 탐지기술, 정보보호학회지, 제17권, 제4호, pp.74-82, August 2007.
[7] 김성우, 해킹/파괴의 광학(개정판), 와이미디어, 2006.

저 자 소 개

홍 성 혁(Sunghyuck Hong) [정회원]

- 1995년 2월 : 명지대학교 컴퓨터공학과 (학사)
- 2007년 8월 : Texas Tech University, Computer Science (Ph.D)
- 2007년 9월 ~ 2012년 2월 : Senior Programmer, Texas Tech University, Office of International Affairs
- 2012년 3월 ~ 현재 : 백석대학교 정보통신학부 교수

<관심분야> : 네트워크 보안, 해킹, 센서네트워크