윈도우 서버의 계정관리방식

1) 워크그룹 방식
- 각각의 계정과 자원을 시스템별로 관리하는 방식으로 소규모 네트워크에 적합하다
- 피어 투 피어 라고도 하며 전용 서버 없이 모든 시스템이 서버이면서 클라이언트 기능을 가지며 서로 동등하다
- 서버 관리자가 필요없으며 보안 관련 정보는 각 시스템의 로컬 디렉터리 데이스(SAM DB)에 의해 제공된다
- Active Directory가 구축되지 않은 상태로서 다른 시스템을 접근할 때 수시로 액세스에 필요한 사용자 계정과 암호를 요구한다.
- Active Directory가 구축되지 않기에 로컬사용자 계정의 저장 위치 = %SystemRoot%\System32%\config\SAM

2) 도메인 방식
- 모든 계정과 자원을 특정 서버에서 관리하는 중앙 집중식 방식이다.
- 사용자에게 적절한 사용 권한을 설정하면 사용자는 다른 컴퓨터에 자원을 지정한 권한대로 접근할 수 있다
- Active Directory가 구축된 상태에서 가능하며 기존의 Window NT기반의 도메인보다 확장된 기능을 제공한다
- 로컬 사용자 계정은 Active Directory에 저장이 됨.

윈도우 시스템 이벤트 로그 파일명

1) 윈도우 XP 이하 버전 -> evt확장자
  - 애플리케이션 로그 : %SystemRoot%\System32\Config\AppEvent.Evt
  - 시스템 로그 : %SystemRoot%\System32\Config\SysEvent.Evt
  - 보안 로그 :%SystemRoot%\System32\Config\SecEvent.Evt

2) 윈도우 Vista 이상 버전 -> evtx확장자
  - 애플리케이션 로그 : %SystemRoot%\System32\winevt\Logs\application.evtx
  - 시스템 로그 : %SystemRoot%\System32\winevt\Logs\system.evtx
  - 보안 로그 : %SystemRoot%\System32\winevt\Logs\security.evtx

윈도우 운영체제 감사 정책

1) 개체 액세스 : 파일, 디렉터리, 레지스트리, 프린터 등의 객체에 대한 접근 성공/실패여부를 기록할지를 결정

2) 계정 관리 : 사용자 계정 또는 그룹의 생성, 변경, 삭제, 암호의 설정 및 변경 등의 이벤트 성공/실패 로그를 기록

3) 계정 로그온 이벤트 : 도메인 계정에 대한 로그온 성공/실패 관련 이벤트 로그를 기록할지를 결정

4) 권한 사용 : 권한 사용의 성공 및 실패를 감사할 경우 사용자 권한을 이용하여 할때마다 이벤트 생성

5) 디렉터리 서비스 액세스 : Active Directory 개체의 시스템 액세스 컨트롤 목록에 나열된 사용자가 해당 개체에 액세스를 시도할 때 이벤트 생성

6) 로그온 이벤트 : 로컬 계정에 대한 로그온/오프 성공/실패에 대한 이벤트를 기록할지를 결정

7) 시스템 이벤트 : 시스템 시작 또는 종료, 보안 로그에 영향을 미치는 이벤트 등을 감사할지를 결정

8) 정책 변경 : 감사 정책 변경의 성공 및 실패를 감사

트를 감사할지를 결정
기본 로그
1) 응용프로그램 로그 : 응용 프로그램이 남기는 다양한 이벤트가 저장 / 어떤 상황에 어떤 로그를 남길지는 해당 응용프로그램을 개발한 개발사에 의해 결정
길지는 감사 로그 설정에 의해 결정
를 저장
보안 운영체제 -> 기존의 운영체제 내에 보안 커널을 추가로 이식한 운영체제
1) 보안 커널 : 주체, 객체 간의 모든 접근과 기능을 중재하는 보안 정차를 구현한 하드웨어, 펌웨어, 소프트웨어 등을 말함
2) 참조 모니터 - 보안 커널의 가장 중요한 부분으로 주체, 객체 간의 접근통제 기능을 수행하는 핵심 모듈 - 주체와 객체 사이의 정보흐름을 감사하는 보안모듈로 보안 커널 데이터베이스(SKDB)를 참조하여 보안 정책을 시행 - 일반적인 구현은 주체와 객체 사이에 정보 흐름의 통로가 되는 시스템 콜을 감시하는 것으로 운영체제의 커널과 독립적으로 동작할 수 있도록 모듈 형태로 구현 (요구사항) - 반드시 부정 조작이 없어야한다 - 항상 무시되지 않고 호출되어야 한다 - 모든 동작에 대해서는 항상 분석과 테스트를 통해 확인될 수 있어야 한다
취약점
널 세션 취약점 : 윈도우가 설치된 네트워크의 다른 원격 컴퓨터에 사용자명과 패스워드를 널로 해서 접속할 수 있게 해주는 것을 말한다. -> 네트워크에 연결된 윈도우 시스템 간에 아이디/패스워드 없이 다른 시스템에 접속할 수 있는 취약점
치한 실행파일(명령)에 의해 의도하지 않은 명령이 실행될 수 있는 취약점 -> PATH 환경변수의 '.'(현재 디렉터리 지정) 설정을 맨 뒤로 이동시키거나 불필요한 경우 삭제한다.
UNIX/Linux 시스템의 디렉터리 구조
1) / : root 디렉터리나 최상위 디렉터리
2) /etc : 시스템 설정 파일이 저장되는 디렉터리
3) /dev : 특수 파일(장치 파일)이 저장되는 디렉터리
4) /usr/bin : 디폴트 사용자 명령어가 저장되는 디렉터리
5) /usr/include : C언어 라이브러리 헤더 파일이 저장되는 디렉터리
6) /usr/lib : C언어 라이브러리가 저장되는 디렉터리
7) /usr/sbin : 시스템 관리 명령어가 저장되는 디렉터리
8) /home : 사용자 홈 디렉터리가 저장되는 디렉터리

9) /tmp : 임시 파일이 저장되는 디렉터리

10) /var : 시스템 로그가 저장되는 디렉터리

리눅스 서버의 프로세스관리를 위한 명령어

top 명령어 : CPU 정보뿐만 아니라 물리적인 메모리, 스왑, 개별 프로세스 정보 등 시스템 자원의 다양한 정보를 실시간으로 모니터링할 수 있도록 종합적으로 보여준다
: 시스템의 전체적인 운영상황을 모니터링할 수 있는 명령어

pstree 명령어 : 현재 실행 중인 프로세스들의 부모-자식 관계를 트리구조로 보여주는 명령어로 프로세스의 상호관계를 파악할 때 유용하다

<프로세스 우선권 순위 설정 명령>
- 우선권 순위는 -20 ~ 19까지 설정하며 작을수록 우선권이 높다.
- nice 명령어 : 프로세스의 우선권 순위를 설정하여 실행하는 명령어
- renice 명령어 : 현재 실행 중인 프로세스의 우선권 순위를 변경하는 명령어

find 명령어 :
type옵션 -> 정규 파일 : f ,
접근권한 옵션 -> -perm [-]mode : -가 없으면 정확히 mode권한과 일치하는 파일, -가 있으면 mode권한을 포함하는 파일을 검색
파일 시간 속성에 따라 -atime(access time) : 파일을 마지막으로 접근한 시간을 지정
-mtime(modification time) : 파일의 내용이 마지막으로 변경된 시간을 지정
-ctime(change time) : 파일의 속성이 마지막으로 변경된 시간을 지정
기간 관련 -> +-가 없으면 정확히 n일, +n이면 n일 초과, -n이면 n일 미만 인 파일을 검색

tail 명령어 : 리스트의 아래부터 출력 -> -f 옵션을 사용하면 파일의 내용을 실시간으로 확인할 수 있음

ln 명령 : link 시키는 명령어 / -s 옵션이 있으면 심볼릭 링크로, 없으면 하드링크 / 하드링크는 파일에만 링크하고, 심볼릭 링크는 파일 또는 디렉터리에 링크할 수 있다

리눅스 시스템에서 컴파일

1) 정적 링크(Static Link)
- 컴파일 시 정적 라이브러리(/usr/lib 디렉터리에 있는 .a파일)을 사용하여 실행 파일 내에 라이브러리 함수가 모두 포함되도록 링크하는 방식

2) 동적 링크(Dynamic Link)
- 컴파일 시 공유 라이브러리(/usr/lib 디렉터리에 있는 .so 파일)를 사용하여 프로그램 실행 시에 외부 공유 라이브러리 함수를 동적으로 링크하는 방식
- 동적 링크 방식으로 컴파일된 실행 파일은 실행 시 외부 공유 라이브러리에 있는 함수 주소를 찾아내는 과정이 필요 이때, PLT와 GOT테이블을 참조
- PLT(Procedure Linkage Table) : 외부 공우 라이브러리 함수를 사용할 수 있도록 주소를 연결해 주는 테이블
- GOT(Global Offset Table) : PLT에서 호출하는 resolve() 함수를 통해 찾아낸 함수의 실제 주소가 저장되어 있는 테이블

*링크 : 프로그램 내에서 라이브러리에 있는 함수를 호출할 깨 호출된 함수와 라이브러리에 있는 함수 코드를 연결해 주는 과정

crontab파일

crontab -l : 자신의 crontab파일에 예약된 작업을 확인하기 위한 명령어

crontab파일을 조작하기 위한 명령어
(리눅스)
crontab -u 계정명 -l : crontab파일 출력
crontab -u 계정명 -e : crontab파일 편집
crontab -u 계정명 -r : crontab파일 삭제
(유닉스)
crontab -l 계정명 : crontab파일 출력
crontab -e 계정명 : crontab파일 편집
crontab -r 계정명 : crontab파일 삭제

버퍼 오버플로우 공격의 대응 및 예방책

- 입력값의 크기를 제한하는 strncpy() 함수를 사용하거나 코딩시 입력값의 크기를 검사(ex. Strlen(argv[1]) > sizeof(buffer)|1024 일경우 진행x도록 설정) *argv[0] : 프로그램명, argv[1] : 첫번째 매개변수

는 것 == 카나리 단어 기법

3) 스택쉴드 : 함수 시작 시 리턴 주소를 Global RET라는 특수 스택에 저장해 두었다가 함수 종료 시 저장된 값과 스택의 RET 값을 비교해 다를 경우 프로그램을 종료

- 메모리 공격을 방어하기 위해 주소 공간 배치를 난수화하는 기법으로 실행 시마다 메모리 주소를 변경시켜(== 스택, 힙, 라이브러리 등의 배치를 랜덤 -> 매번 다른 주소에서 실행) 악성코드에 의한 특정 주소 호출을 방지
(설정) randomize_va_space 커널 파라미터를 0으로 설정하면 사용하지않는다는 의미 / 1로 설정하면 힙 이외는 모두 랜덤하게 설정한다는 의미 / 2로 설정하면 모두 랜

리눅스 패스워드 설정 명령어

passwd -l <계정명> : 패스워드 잠금 설정 -> 서버로 원격 접속을 못하게 됨

passwd -u <계정명> : 패스워드 잠금해제

리눅스 패스워드 설정 파일 : /etc/login.defs --- 권장 설정

PASS_MIN_LEN 8 : 패스워드 최소 길이를 8자 이상으로 설정한다

PASS_MAX_DAYS 90 : 패스워드 최대 사용기간을 90일로 설정한다

PASS_MIN_DAYS 1 : 패스워드 최소 사용기간을 1일로 설정한다

유닉스 시스템의 핵심 컴포넌트

1) 커널
- 유닉스 운영체제의 핵심으로, 메인 메모리에 상주하여 컴퓨터 자원을 관리한다
- 하드웨어 특성으로부터 프로그램들을 격리하고, 하드웨어와 직접 상호 작동함으로써 프로그램들에 일관된 서비스를 제공
(기능)
- 기본적으로 프로세스와 파일의 관리
- 입출력장치 관리, 메모리 관리 및 시스템호출 인터페이스
- 하드웨어나 유틸리티 또는 응용 프로그램들은 정의된 시스템 호출을 통해서 커널과 통신
- 부팅시 가장 먼저 로드되는 운영체제의 핵심 부분으로 주기억장치에 상주하여 프로세스 스케줄링, 기억 장치 관리, 파일시스템 관리, 운영체제의 고유 기능을 제공한다

2) 쉘 : 커널과 사용자 간의 인터페이스를 담당하며, 사용자 명령의 입출력을 수행하며, 프로그램을 실행시킨다
: 쉘은 이용자와 시스템과의 대화를 가능하게 해주며, 이용자가 입력한 문장을 읽어 그 문장을 요청하는 시스템 기능을 수행하도록 해주는 명령 해석기

3) 파일 시스템 : 디렉터리, 서브 디렉터리, 파일 등의 계층적인 트리구조를 제공한다

유닉스 파일시스템 구성

1) 부트 블록 : 운영체제를 부팅하거나 초기화하기 위한 부트스트랩 코드를 담고 있는 블록

2) 슈퍼 블록 : 해당 파일시스템을 관리하기 위한 정보를 담고 있는 블록

3) 아이노드 블록 : 해당 파일시스템의 파일들에 대한 속성 정보를 담고 있는 아이노드 구조체 리스트

4) 데이터 블록 : 실제 파일의 내용(데이터)이 저장되는 블록

유닉스 관련 로그 파일

utmp(x) : 시스템에 현재 로그인한 사용자들에 대한 상태를 기록 / 'who', 'w', 'finger'명령으로 그 내용을 볼 수 있음

/var/log/wtmp : 사용자가 로그인 또는 로그아웃할 때마다 그 정보가 기록 / 'last' 명령으로 내용 확인 + last 계정명 : 해당 계정의 로그인/로그아웃 기록 확인

/var/adm/loginlog : 로그파일에 5번 이상 실패한 로그인 정보를 기록 -> 텍스트 형식이기에 vi등 편집기를 통해 확인가능
cf) /var/log/btmp : 실패한 로그인 시도에 대한 기록을 담고 있는 로그파일(리눅스) / 'lastb' 명령으로 내용 확인

acct/pacct : 사용자들에 의해 실행된 모든 명령이 기록 /'lastcomm' 명령으로 내용 확인 cf) 리눅스 시스템은 'acctcom' 명령어로 확인 가능

해당 일수 이내의 접속한 기록 확인

/var/adm/sulog : 'su' 명령을 사용한 결과를 저장한 파일로 SunOS를 포함한 Unix 계열에서만 볼 수 있음
cf) 리눅스 계열의 경우 /var/log/secure 로그파일에 'su' 명령을 사용한 결과가 남음

/var/adm/loginlog : 5번 이상 실패한 로그인 시도에 대한 기록으로 텍스트 형식으므로 vi 등의 편집기를 통해 로그 내용을 확인할 수 있다
cf) 리눅스 계열의 경우 /var/log/btmp 로그파일에 실패한 로그인 시도에 대한 기록을 남기며 'lastb'명령을 통해 그 내용을 확인가능

리눅스 로그 파일 형식

ID) (프로세스가 생성한 메시지)

/var/log/boot.log : 리눅스가 부팅될 때 파일시스템에 대한 체크와 서비스 데문들의 실행 상태를 기록하고 있는 로그파일로 성공/실패 여부를 확인할 수 있다.

/var/log/xferlog : FTP 로그파일로 proftpd, vsftpd 등 FTP 데몬의 서비스 내역을 기록 / FTP로 로그인하는 사용자에 대한 내역, 파일을 업로드/다운로드한 내역을 기록

syslog

분산 시스템 또는 프로세스의 로그(메시지)를 중앙 수집기로 보내 로깅과 분석을 수행하가 위한 UDP 기반의 인터넷 표준 프로토콜
기밀성, 무경성, 가용성 등 정보보호 특성을 고려하지않고 개발됨
즉, 공격자는 UDP를 통해 로그를 전송할 때 syslog 메시지를 모니터링하여 중요 정보를 알아낼 수 있다.

syslogd 데몬 프로세스를 원격 로그 서버로 설정 시 514/udp(syslog 기본 포트)를 사용

BEEP : the Blocks Extensible Exchange Protocol)는 연결 지향적이고, 비동기적인 연결을 위한 응용 프로그램 프로토콜 프레임 워크로 내부적으로 인증, 프라이버시, 재전송을 통한 신뢰성 등을 보장하고 있으므로 좀 더 안전하고 신뢰성 있는 syslog 메시지 전달이 가능하다
syslog 메시지 전송 시 기밀성 보장을 위해 syslog 서버와 로그 수집 대상 서버의 IP를 제외한 페이로드를 보호할 수 있음

syslog.conf 설정 형식

리눅스 시스템의 기본적인 로그 파일은 syslogd에 의해 제어 & 설정파일인 syslog.conf로 로그를 기록시 파일들의 저장위치와 파일명, 로그 레벨 등의 변경이 가능

facility.priority; facility.priority; …..... Action(logfile-location)

- priority : 로그 수준을 지정한 수준 이상의 상황이 발생했을 때 로그를 남긴다
- action : 로그 위치를 지정하는 필드로 로그 파일 명을 지정하거나 콘솔 또는 원격 로그 서버를 지정할 수 있음 , 원격 로그 서버를 지정할 경우에는 '@원격 로그 서버 IP' 형식으로 지정

<로그 수준(level) 유형> ( )는 단축형태
1) Emergency(emerg) : 시스템이 전면 중단되는 패닉 상태 / 전체 공지가 필요한 상황
2) alert(alert) : 즉각적인 조치가 필요한 상황
3) Critical(crit) : 하드웨어 등의 심각한 오류가 발생한 상황
4) Error(err0 : 일반적인 에러/오류가 발생한 상황
5) Warning(warning) : 경고(주의를 요구하는) 상황
6) Notice(notice) : 에러/오류는 아니지만, 의미있는 이벤트가 발생하여 관리자의 조치가 필요한 상황
7) Information(info) : 정보 메시지
8) debug(debug) : 디버깅용 메시지

SSH 보안 조치(/etc/profile에 설정) *etc/profile은 모든 사용자의 로그인 시 적용되는 환경 설정 파일

export TMOUT = ( ); : SSH Login 세션 타임아웃을 ( )초로 모든 사용자에게 적용

xinetd의 설정

disable = yes/no : 활성화 비활성화 여부의 결정

only_from : 접근을 허용할 특정 IP 주소 또는 IP주소 대역(주소대역 표시는 CIDR표기)

no_access : 전근을 차단할 특정 IP 주소 또는 IP주소 대역을 설정

access_times : 접근을 허용할 시간(24시간 기준) 범위를 설정

cps = [개수] [시간] : 개수는 초당 최대 연결개수를 의미, 시간은 초당 최대 연결개수 초과 시 연결을 제한하는 시간(초 단위)를 의미

instances : 동시에 서비스할 수 있는 서버(서비스 프로세스) 개수를 제한

per_source : 출발지 IP별 최대 연결개수를 설정

PAM 모듈 설정(사용자 계정의 인증 관련) - /etc/pam.d/system-auth(서비스 설정 파일)

PAM(Pluggable Authenrication Module) : 사용자를 인증하고 그 사용자의 서비스에 대한 액세스를 제어라는 모듈화된 방법

<모듈 유형>type
- auth : 사용자 계정의 비밀번호(인증 정보) 검증 등 사용자 신원확인을 수행하는 유형
- account : 사용자 계정의 유효성(계정 유효기간, 접근 허용 여부 등을 검증하는 유형
- password : 사용자 계정의 비밀번호(인증 정보) 설정 및 변경 조건을 지정하는 유형
- session : 사용자 계정 인증 처리 전후에 수행할 작업을 지정하는 유형

<제어 방식>control : 모듈의 실행 결과에 따라 어떤 동작을 해야하는지 결정
- requisite : (필수 모듈) 인증 실패 시 즉시 인증을 거부
- required : (필요 모듈) 인증에 실패해도 나머지 모든 모듈을 수행한 후 최종적으로 인증을 거부
- sufficient : (충분 모듈) 선행 모듈이 실패해도 현재 모듈 인증에 성공하면 최종적으로 인증을 허용한다
- optional : (옵션 모듈) 인증 성공/실패 결과는 모두 무시한다
- include : 다른 PAM 설정 파일을 포함

<인수>
- deny = (_) : ( )회 입력 실패시 패스워드 잠금
- unlock_time = (_) : 계정이 잠긴 후 마지막 계정 실패 시간부터 120초가 지나면 자동 계정 잠금 해제
- no_magic_root : root 계정은 패스워드 잠금 설정을 적용하지 않음
- reset : 접속 시도 성공시 실패 횟수 초기화

레드햇 리눅스 시스템에서 비밀번호 복잡성 설정시 사용하는 변수명 -> /etc/pam.d/system-auth설정

1) lcredit=-1 : 최소 소문자 요구 -> (권장) 최소 1자 이상의 소문자 요구

2) ucredit=-1 : 최소 대문자 요구 -> (권장) 최소 1자 이상의 대문자 요구

3) dcredit=-1 : 최소 숫자 요구 -> (권장) 최소 1자 이상의 숫자 요구

4) ocrediit=-1 : (other character) 최소 특수 문자 요구 -> (권장) 최소 1자 이상의 특수문자 요구

5) minlen=8 : 최소 패스워드 길이 요구 ->(권장) 최소 8자리 이상 요구

6) difok=N : 기존 패스워드와 비교 -> 새 비밀번호가 이전 비밀번호와 달라야 하는 문자 수(N)

RPC(Remote Procedure Call)

- 분산 환경에서 원격 서버 응용프로그램의 함수나 프로시저를 호출해 주는 프로세스 간 통신 기술
- 별도의 원격 제어를 위한 코딩 없이 다른 주소 공간에서 함수나 프로시저를 실행할 수 있도록 하는 프로세스 간 통신 기술
- 버퍼 오버플로루, 도스, 원격 실행 등의 취약성이 존재하는 RPC 서비스를 통해 비인가자의 root권한 획득 및 침해사고 발생 위험이 있으므로 불필요한 RPC 서비스를 중지해야 한다

<불필요한 RPC의 종류>
- rstatd : CPU와 가상메모리 사용 통계, 네트워크 가동 시간, 하드디스크에 대한 정보를 제공하는 데몬
- wall : 메시지를 네트워크의 모든 사용자에게 전송하는 요청을 처리하는 데몬
- sadmind : 원격에서 시스템을 관리하거나 모니터링을 쉽게 도와주는 데몬
- rexd : 원격에서 서버 명령어를 실행하도록 하는 데몬
- ruserd : 현재 네트워크에 있는 사용자 리스트를 반환해 주는 데몬

그외 rpc.statd, rpc.ttdbserverd, rpcypipdated, sprayd, rpc.nisd, rpc.pcnfsd등이 존재

Logrotate

서비스

기본 설정 파일 : logrotate.conf / logrotate를 적용할 개별 프로세스의 설정 파일 : /etc/logrotate.d / 주기적으로 logrotate를 실행하기 위한 서비스 : cron

<옵션>
- daily, weekly, monthly, yearly : 매일, 매주, 매월, 매년 단위로 로그파일 순환
- rotate [개수] : 개수만큼 로그 순환 파일을 사용
- compress, no compress : 로그파일을 압축하여, 압축 없이 보관
- missingok : 로그파일이 없어도 오류를 발생시키지 않는다
- notifempty : 'not if empty'의 의미로 로그파일이 비어있는 경우 순환하지 않음
- sharedscripts : 로그파일이 여러 개 있어도 스크립트를 공유하여 prerotate, postrotate 스크립트를 한 번만 실행
- postrotae/endscript : 순환 후 스크립트 파일 실행
- size [크기] : 로그파일이 크기가 되면 순환한다
- create : 오래된 로그부터 순환한 후 새롭게 로그파일을 생성한다.
- dateext : 로그파일의 확장자로 날짜를 붙여서 저장한다

리눅스 시스템에서 루트 계정의 원격 호스트 접속 차단을 위한 보안 설정

1) securetty 파일(etc/securetty)에 가상 터미널(pts/번호) 설정을 제거하거나 주석 처리 -> Talnet서비스를 통해 원격 호스트에 루트 계정으로 직접 접근하는 것을 차단

2) SSH 데몬(sshd) 설정 파일인 sshd_config 파일의 PermitRootLogin 옵션을 no로 설정 -> SSH 서비스를 통해 원격 호스트에 root계정으로 직접 접근하는 것을 차단

한 일반 사용자 계정과 실행을 허용할 명령어를 설정
->1),2)를 설정해도 일반 사용자가 원격 접속한 이류에 su명령을 통해 root 쉘을 실행가능 -> root쉘로 전환하는 것을 봉쇄하고 루트 권한이 필요한 명령어에 대해서는

포멧 스트링 취약점

링 취약점이 존재 -> 공격자는 목표 애플리케이션에 대한 통제권을 장악가능

<보안 위협>
- 프로세스 공격(종료/다운)
- 프로세스의 메모리 읽기(확인)
- 프로세스의 메모리 쓰기(변조)를 악용한 임의 코드 실행

rsync 동기화 프로그램

서버 간 또는 서버 내 디렉터리 및 파일을 전송하고 동기화 할 수 있는 프로그램
추가된 파일이나 수정된 파일만 동기화하거나 일부 파일이나 디렉터리를 제외하고 동기화할 수 있다.
-> 잘못된 설정은은 서버를 취약한 상태로 만들 수 있음

- path 필드 : 최상위 디렉터리(/)로 설정시 동기화 시 대상 서버의 모든 디렉터리에 접근할 수 있으므로 공격자에 의한 서버의 중요 파일 접근 및 조작이 발생할 수 있다
- uid/gid 필드 : root계정 및 그룹으로 설정하면 동기화시 대상 서버 디렉터리에 root 계정 및 그룹 권한으로 접근할 수 있으므로 공격자에 의한 관리자 권한의 파일 접근 및 조작이 발생할 수 있다

TCP Wrapper(접근제한 설정 도구)

네트워크 서비스에 관련한 프래픽을 제어하고 모니터링 할 수 있는 UNIX 기반의 방화벽도구
cf) iptables : 리눅스 커널 방화벽이 제공하는 테이블들과 그것을 저장하는 체인, 규칙들을 구성할 수 있게 해주는 도구로 규칙 기반의 다양한 패킷 필터일 기능을 제공

inetd.conf 파일에서 해당 서비스의 실행경로를 tcpd로 설정하면 해당 서비스를 접속시 TCP Wrapper의 접근제어를 받게 됨

hosts.allow가 hosts.deny보다 우선

(형식)
설정할 서비스데몬 : IP 주소 (범위일시 -> IP주소/subnetmask) *ALL -> 모든 주소를 의미

운영체제 보안을 위한 분리

1) 물리적 분리 : 사용자별로 별도의 장비만 사용하도록 제한하는 방법으로 강한 형태의 분리가 되지만 현실적/실용적이지 못하다

2) 시간적 분리 : 프로세스가 동일 시간에 하나씩만 실행되도록 하는 방법으로 동시 실행으로 발생되는 보안 문제를 제거

하게 제한된다.

4) 암호적 분리 : 내부에서 사용되는 정보를 외부에서는 알 수 없도록 암호화하는 방법

보안상 취약한 함수와 대체하는 안전한 함수

strcat() -> strcat_s() : 2개의 문자열을 연결시키는 함수

strncat() -> strncat_s() : 2개의 문자열을 지정한 개수만큼 연결시키는 함수

strcpy() -> strcpy_s() : 문자열을 복사해주는 함수

strncpy() -> strncpy_s() : 문자열을 지정한 개수만큼 복사해주는 함수

sprintf() -> sprintf_s() : 서식에 따른 문자열을 문자 배열(버퍼)에 출력하는 함수
gets() -> gets_s() or fgets() : 표준 입력에서 문자열을 입력받는 함수
scanf() -> scanf_s() : 표준입력에서 서식에 따라 데이터를 입력받는 함수
운영체제별 패스워드의 최대 사용기간 정책 권장 설정 확인
1) SunOS (최대 12주 이하) cat /etc/defualt/passwd MAXWEEKS=12
2) LINUX (최대 90일 이하) cat /etc/login.defs PASS_MAX_DAYS 90
3) AIX (최대 12주 이하) cat /etc/security/user maxage=12
4) HP-UX (최대 90일 이하) cat /etc/default/security PASSWORD_MAXDAYS=90
운영체제별 패스워드 최소 길이설정 방법(최소 8자)
1) SunOS cat /etc/defualt/passwd PASSLENGTH=8
2) LINUX cat /etc/login.defs PASS_MIN_LEN 8
3) AIX cat /etc/security/user minlen=8
4) HP-UX cat /etc/default/security MIN_PASSWORD_LENGTH=8
OS별 root 계정에 대한 원격 접속 제한 설정

1) SunOS 보안 설정
- /etc/default/login 파일의 CONSOLE라인을 통해 root의 원격 접속 제한 if) 주석처리시 외부에서 root계정 접속가능 / 주석처리x시 콘솔에서만 접근 가능

2) AIX 보안 설정
- /etc/security/user 파일의 rlogin 설정을 false일시 원격에서 접근할 수 없음

3) Liinux 보안 설정
- 로그인 서비스에 pam_securitty.so PAM 모듈을 추가한 후 /etc/securetty파일에 'pts/~'터미널을 모두 제거(또는 주석 처리)한다

출력 관련 (출력 방향 재지정과 표준에러 출력)

'>' : 출력 파일 존재 시 그 내용을 지우고 새롭게 출력

'>>' : 출력 파일 존재 시 기존 내용에 추가하여 출력

1' : STDOUT(Standard output)을 의미하는 파일 디스크립터 -> 표준출력(정상 결과)

'2' : STDERR(Standard error)을 의미하는 파일 디스크립터 -> 표준에러

xferlog 정보

(인증된 사용자 ID) (완료 상태)

(전송파일 유형)
a : ascii(아스키, 텍스트) , b : binary(바이너리)

(액션 플래그)
_ : 액션 없음 , C : 파일이 압축됨, U : 압축된 파일이 해제됨, T : 파일이 tar로 묵음(아카이브됨)

(전송 방향)
i : incoming(서버로 파일 업로드), o : outgoing(서버에서 파일 다운로드), d : delete(서버파일 삭제)

r : real(로컬 시스템 계정으로 접근), a : anonymous(익명 계정으로 접근), g : guest(게스트 계정으로 접근) *게스트 계정이란 시스템 계정이 아닌 FTP 접속만을 위해 FTP 서버에서 만든 가상의 계정을 의미

(인증 방법)
0 : 없음, 1: RFC 931인증

(완료 상태)
c : complete(전송 성공), i : incomplete(전송 실패)

저널링

Ext2 파일시스템에서 사용하는 fsck의 시간이 오래 걸리는 단점을 보완한 파일시스템 복구 기술

복구 시간을 단축하기 위해 데이터를 디스크에 쓰기 전에 로그에 데이터를 남겨 시스템의 비정상 종료 시에도 로그를 사용해 fsck보다 빠르고 안정적인 복구 기능을 제공

Ext2 파일시스템에서는 파일시스템을 복구하기 위해 슈퍼블록, 아이노드 등을 모두 검사해야 했지만, 저널링 기술이 도입된 Ext3 파일시스템의 경우에는 파일을 실제로 수정하기 전에 우선 로그에 수정된 내용을 저장하기 때문에 로그만을 검사하여 속도와 복구 안정성을 향상시킨다

레이스 컨디션 공격
<동작 원리> 공격자는 임시파일을 만들어 놓고 피해 프로그램에서 한 파일을 생성하기 전에 동일한 이름으로 심볼릭 링크를 걸게 되면 피해 프로그램은 공격자가 만들어 놓은 파일을 사용하게 되어 링크 걸린 임시파일에 내용이 기록되고 한 파일를 지우더라도 임시파일의 내용은 남게됨 이 때 피해프로그램에서 파일을 먼저 생성하면 공격은 실패하므로 먼저 생성하는 시점을 확보하기 위해 링크를 반복함 -> 많은 반복과 경재이 필요함
<대응 방안> - 가능하면 임시파일을 생성하지 않는나 - 파일 생성 시 이미 동일한 파일이 존재하는 경우 파일 생성 또는 쓰기를 금지한다 - 사용하고자 하는 파일에 링크가 걸려있이면 실행을 중단한다 - umask를 최하 022 정도로 유지하여 임시로 생성한 파일이 공격자에 의해 악의적으로 삭제되지 않도록 한다.
인터럽트
의 변경을 말함
그램
PCB(프로세스 제어 블록)
OS가 프로세스를 관리하는데 필요한 모든 정보를 유지하는 자료구조 테이블
프로세스 디스크립터라고 하며, 프로세스가 생성할 때 만들어지며, 모든 프로세스는 각각 고유한 프로세스 디스크립터를 가진다
한다
수행이 완료된 프로세스는 해당 PCB도 함께 삭제된다, PCB의 내용은 프로세스의 상태 변화가 일어났을 때 프로세스 관리자가 그 내용을 변경한다
메모리의 버퍼 할당 영역(변수 할당 공간)
1) 스택 영역 - 함수 처리를 위한 지역변수 및 매개변수가 위치하는 메모리 영역 - 스택에 할당된 버퍼(변수)를 오버플로우시켜서 스택의 복귀주소 영역을 변조시키면 공격자가 원하는 임의의 코드를 실행시킬 수 있다 -> 스택 오버 플로우 공격
- 프로그래머가 필요시 할당하고 해제할 수 있는 동적 메모리 영역 - 힙영영에 할당된 버퍼 크기를 초과하는 양의 데이터(실행 코드)를 입력하여 메모리의 데이터와 함수 주소 등을 변경하여 공격자가 원하는 임의의 코드를 실행시킬 수 있다 -> 힙 오버 플로우 공격 - 힙은 일반적으로 연결 리스트 형태로 관리되므로 스택과 달리 모든 버퍼(변수)가 연속된 메모리 공간에 할당되는 것이 아니다 - 즉, 소스코드 및 메모리 분석을 하더라도 변수의 기능이나 주소를 쉽게 예측하기 어려워 상대적으로 버퍼 오버플로우 빈도는 낮다
net user 명령어
윈도우 시스템에서 로컬 계정을 추가/수정/삭제하거나 로컬 계정 정보를 표시하는 명령어

<형식>
net user : 전체 로컬 계정 정보 확임
net user <계정명> : 해당 계정의 속성 확인
net user <계정명> /active:yes : 해당 계정 사용함(활성 상태)
net user <계정명> /active:no : 해당 계정 사용 안 함(비활성 상태)

용어

이벤트 뷰어 : 윈도우 OS에서 로그를 조회하고 관리하는 도구

스왑 공간(Swap space) : 디스크(보조기억장치)의 일정 영역으로 메모리가 부족할 경우 마치 메모리처럼 사용하는 공간

의 공격
: 원격 서버나 원격 서비스에 접속할 때 사용자의 실제 패스워드를 모르는 상태에서도 탈취한 사용자의 패스워드 해시값을 이용하여 인증을 시도한다

미미카츠 : 윈도우 시스템에서 사용자 계정, 패스워드 등의 자격증명 정보를 수집할 수 있는 도구

감사 정책 : 윈도우 로그 관리와 관련해서 서버 관리자가 가장 먼저 고려해야할 대상
: 어떤 로그를 남길지 정의한 규칙 / 해당 정책에 의해 지정한 이벤트 범주에 대해서만 로그가 남음
: 설정 파일 -> secplo.msc

에 방문한 후 무작위로 대입하는 공격방식
: 편의를 위해 한 가지 ID와 비밀번호를 여러 시스템/사이트에서 사용하는 사용자의 취약성을 이용한 공격

net share "공유 폴더(디렉터리) 이름" /delete : 공유된 폴더를 해제하는 명령어

문맥 : 특정 프로세스와 관련된 정보들의 총집합을 의미 / 하나 프로세스의 문맥은 그 프로세스의 프로세스 제어 블록(PCB)에 기록한다.

재하는 작업

상호 배제
- 둘 이상의 프로세스 또는 스레드가 동일한 자원에 동시에 접근하는 것을 차단하기 위한 동기화기법(자원의 순서를 제어하는 기법)
- 프로세스가 특정 자원을 사용하려 할 때 락을 획득한 후 사용하고 사용이 끝나면 락을 해제하여 다른 프로세스가 사용할 수 있도록 하는 방식
- 접근 순서에 따라 프로세스 간에 서로 락이 걸린 자원을 무한정 기다리는 교착상태가 발생할 수 있는 문제가 있다

- 윈도우 시스템에서 자체적으로 제공하는 하나 이상의 볼륨(드라이브)을 암호화하는 기능으로 TPM(신뢰할 수 있는 플랫폼 모듈)을 사용하여 초기 시작 구성요소의 무결성을 검사하는 암호화 방식
*TPM : 하드웨어와 소프트웨어, 펌웨어 인증을 검사하는 전용칩 / 승인 없는 변경을 감지했을 경우 pc는 제한된 모드로 부팅되어 잠재적인 공격자의 악의적인 행위를 차