정보보안론 7주차 강의

! 시험 – 시스템 메모리의 구조 !

프로그램을 동작시키면 프로그램이 동작하기 위한 가상의 공간이 메모리에 생성.

메모리 공간은 목적에 따라 상위 메모리와 하위 메모리로 나뉨

상위 메모리에는 스택, 하위 메모리에는 힙 생성

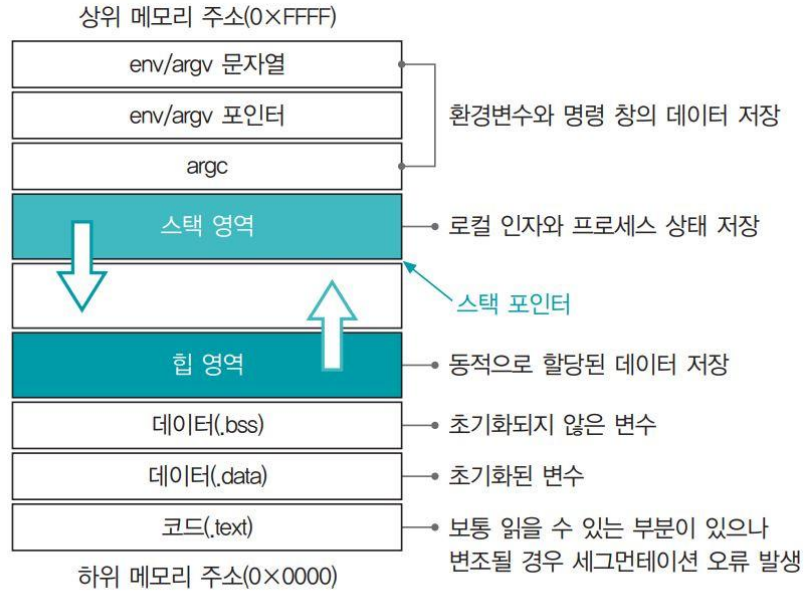

그림 5-2 메모리의 기본 구조

! 시험 – 스택 / 힙 영역, 레지스터 !

스택 영역 : 프로그램 로직이 동작하기 위한 인자와 프로세스 상태를 저장하는 데 사용.

힙 영역 : 프로그램이 동작할 때 필요한 데이터 정보를 임시로 저장하는 데 사용

레지스터 : CPU의 임시 메모리로 CPU 연산과 어셈블리어의 동작에 필요

표 5-1 80×86 CPU의 레지스터

범주	80386 레지스터	이름	비트	용도
범용 레지스터	EAX	누산기(accumulator)	32	산술 연산에 사용(함수의 결과 값 저장)
	EBX	베이스 레지스터(base register)	32	특정 주소 저장(주소 지정을 확대하기 위한 인덱스로 사용)
	ECX	카운트 레지스터(count register)	32	반복적으로 실행되는 특정 명령에 사용(루프의 반복 횟수나 좌우 방향 시프트 비트 수 기억)
	EDX	데이터 레지스터(data register)	32	일반 데이터 저장(입출력 동작에 사용)
세그먼트 레지스터	CS	코드 세그먼트 레지스터 (code segment register)	16	실행 기계 명령어가 저장된 메모리 주소 지정
	DS	데이터 세그먼트 레지스터 (data segment register)	16	프로그램에서 정의된 데이터, 상수, 작업 영역의 메모리 주소 지정
	SS	스택 세그먼트 레지스터 (stack segment register)	16	프로그램이 임시로 저장할 필요가 있거나 사용자의 피호출 서브루틴이 사용할 데이터와 주소 포함
	ES, FS, GS	엑스트라 세그먼트 레지스터 (extra segment register)	16	문자 연산과 추가 메모리 지정에 사용되는 여분의 레지스터
포인터 레지스터	EBP	베이스 포인터(base pointer)	32	SS 레지스터와 함께 스택 내의 변숫값을 읽는 데 사용
	ESP	스택 포인터(stack pointer)	32	SS 레지스터와 함께 스택의 가장 끝 주소를 가리킴
	EIP	명령 포인터(instruction pointer)	32	다음 명령어의 오프셋(상대 위치 주소)을 저장하며, CS 레지스터와 합쳐져 다음에 수행될 명령의 주소 형성

범주	80386 레지스터	이름	비트	용도
인덱스 레지스터	EDI	목적지 인덱스(destination index)	32	목적지 주소의 값 저장
	ESI	출발지 인덱스(source index)	32	출발지 주소의 값 저장
플래그 레지스터	EFLAGS	플래그 레지스터(flag register)	32	연산 결과 및 시스템 상태와 관련된 여러 가지 플래그 값 저장

! 시험 – 버퍼오버플로 ! 버퍼오버플로가 발생하면 어떤 문제가 발생하는지

버퍼 오버플로 공격 : 데이터의 길이에 대한 불명확한 정의를 악용한 덮어쓰기로 발생

프로그램의 진행방향을 바꿀 수 있다.

! 시험 – 웹보안 ! 웹과 HTTP가 무엇인지, GET/POST 방식

GET 방식

가장 일반적인 HTTP Request 형태로, 요청 데이터의 인수를 웹 브라우저의 URL로 전송

데이터가 주소 입력란에 표시되므로 최소한의 보안도 유지되지 않는 취약한 방식

POST 방식

URL에 요청 데이터를 기록하지 않고 HTTP 헤더에 데이터를 전송.

인숫값을 URL로 전송하지 않으므로 다른 사용자가 링크로 해당 페이지를 볼 수 없음.

! 시험 – 프록시란 !

클라이언트와 서버간의 응답, 요청을 하게 되는데 중계 역할을 하는 서버

! 시험 – 웹의 주요 취약점 !

사용자 데이터를 이용한 인증

❶ 최초 인증 과정은 정상적인 아이디와 패스워드의 입력으로 시작

❷ 패스워드가 올바른 경우의 접속에 대해 인증을 한 뒤 인증 값으로 쿠키와 같은 세션 값을 넘겨줌

❸ 웹 서버가 새로운 페이지에 접근할 때 공격자가 ❷에서 수신한 인증 허용 값을 전달받으면서 해당 세션 이 유효한 인증인지 확인

❹ 공격자는 세션 인증 값은 그대로 사용하고 UserNo 값만 변경하여 다른 계정으로 로그인한 것처럼 웹 서비스를 이용할 수 있음

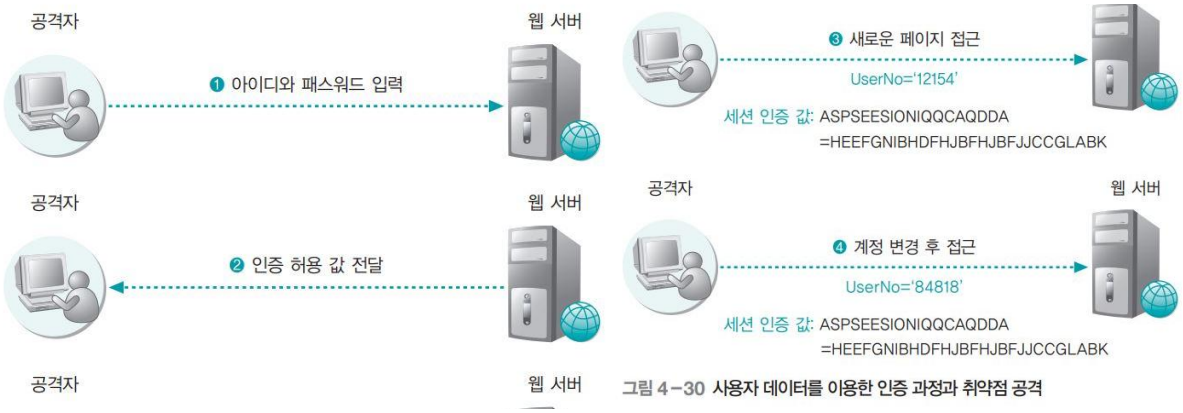

그림 4-30 사용자 데이터를 이용한 인증 과정과 취약점 공격

! 시험 – XSS란, CSRF란, 두개 비교 !

XSS: 공격자가 작성한 스크립트가 다른 사용자에게 전달되는 것

다른 사용자의 웹 브라우저 안에서 적절한 검증 없이 실행되기 때문에 사용자의

세션을탈취하거나 웹 사이트를 변조하고 악의적인 사이트로 이동할 수 있음

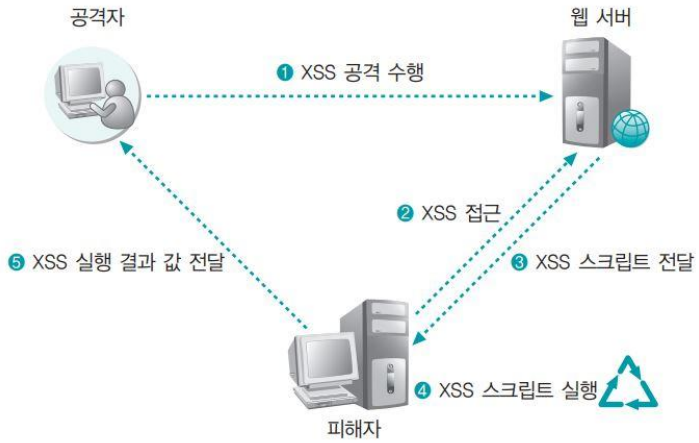

그림 4-31 XSS 공격의 구조

XSS 공격의 구조

❶ 임의의 XSS 취약점이 존재하는 서버에 XSS 코드를 작성하여 저장

❷ 공격자가 작성해놓은 XSS 코드에 해당 웹 서비스 사용자가 접근

❸ 웹 서버는 사용자가 접근한 XSS 코드가 포함된 게시판의 글을 사용자에게 전달

❹ 사용자의 시스템에서 XSS 코드가 실행

❺ XSS 코드가 실행된 결과가 공격자에게 전달되고 공격자는 공격을 종료

CSRF : 특정 사용자를 대상으로 하지 않고 불특정 다수를 대상으로 로그인된 사용자가 자신의 의지와 무관하게 공격자가 의도한 행위(수정, 삭제, 등록, 송금 등) 를 하게 만드는 공격

XSS와 CSRF 비교

기본적으로 XSS 공격과 매우 유사하며, XSS 공격의 발전된 형태로 보기도 함

XSS 공격은 악성 스크립트가 클라이언트에서 실행되는 데 반해 CSRF 공격을 하면 사용자가 악성 스크립트를 서버에 요청

! 시험 – 악성코드, 바이러스, 웜, 토리오 목마, PUP 정의들, 악성 코드 탐지 및 대응책 !

악성코드 : 악의적인 목적으로 개발되어진 소프트웨어 (상대방의 기기를 랜섬웨어로 바꿀

수 있는)

악성코드는 대부분 자가복제 기능을 가진다.

! 밑에 표 시험 !

표 6-1 동작에 의한 악성 코드 분류

악성 코드	설명
바이러스	• 사용자의 컴퓨터(네트워크로 공유된 컴퓨터 포함) 내에서 프로그램이나 실행 가능한 부분을 몰래 변형하여 자신 또는 자신의 변형을 복사하는 프로그램이다. • 가장 큰 특성은 복제와 감염이며, 다른 네트워크의 컴퓨터로 스스로 전파되지는 않는다.
웜	• 인터넷 또는 네트워크를 통해 컴퓨터에서 컴퓨터로 전파되는 악성 프로그램이다. • 윈도우 또는 응용 프로그램의 취약점을 이용하거나 이메일 또는 공유 폴더를 통해 전파되며, 최근에는 공유 프로그램(P2P)을 통해 전파되기도 한다. • 바이러스와 달리 스스로 전파된다.
트로이 목마	• 바이러스나 웜처럼 컴퓨터에 직접적인 피해를 주지는 않지만, 악의적인 공격자가 침투하여 사용자의 컴퓨터를 조종하는 프로그램이다. • 고의적으로 만들어졌다는 점에서 프로그래머의 실수인 버그와는 다르다. • 자기 자신을 다른 파일에 복사하지 않고 인터넷 또는 네트워크를 통해 전파되지 않는다는 점에서 컴퓨터 바이러스나 웜과 구별된다.
PUP	• 잠재적으로 원하지 않는, 즉 불필요한 프로그램이란 의미로, 사용자에게 치명적인 피해를 주지는 않지만 불편함을 주는 악성 코드다. • 프로그램 설치 시 사용자에게 직간접적인 동의를 구하지만 용도를 파악하기 어렵게 한다. • 스파이웨어나 광고가 포함된 악성 코드 제거 프로그램, 웹 사이트 바로가기 생성 프로그램 등이 있다.

! 시험 – 바이러스, 부트바이러스 !

바이러스 : 자기 복제 기능과 데이터 파괴 기능이 있는 프로그램

부트 바이러스 : 플로피디스크나 하드디스크의 부트 섹터에 감염되는 바이러스

MBR과 함께 PC 메모리에 저장되어 부팅 시 자동으로 동작하여 부팅 후에 사용되는 모든 프로그램을 감염

모든 저장장치에 맨 앞에 있는 영역 (0번 섹터) 는 부트섹터이다. 부팅용 저장장치인지 저장용 저장장치인지 구분되어 있다.

! 시험 - 1 세대 원시형 바이러스 !

파일 바이러스 : 파일을 직접 감염시켜 바이러스 코드를 실행시키는 것

파일 바이러스의 감염 위치 : 프로그램을 덮어쓰는 경우 / 프로그램 앞부분에 실행 코드를 붙이는 경우 / 프로그램 뒷부분에 바이러스 코드를 붙이는 경우

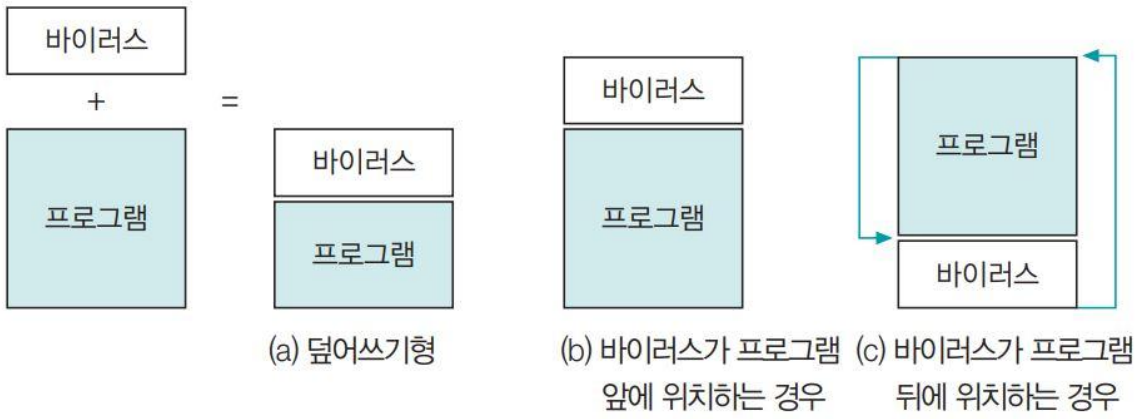

그림 6-4 파일 바이러스의 감염 위치

웜 : 인터넷 또는 네트워크를 통해 컴퓨터에서 컴퓨터로 전파되는 프로그램

트로이 목마 : 악성 루틴이 숨어 있는 프로그램 (숨어있다가 특정한 상황이 되면 악성 코드가 실행 되는 것)

PUP : 사용자에게 직간접적으로 동의를 구하지만 용도를 파악하기 어려운 상태에서 설치되는 프로그램

악성 코드 방지 : 출처를 확인되지 않은 파일, 게시판, 메일은 다운,접근 금지