제 1 장 컴퓨터 보안 개요

금오공과대학교
컴퓨터공학부 컴퓨터공학전공
최태영

컴퓨터시스템보안 목차

보안의 기존개념

- 보안 시스템
  - 가치 있는 자산이 손상되지 않도록 하는 시스템
  - 타인에게 알려지면 곤란한 정보를 외부와 차단은 시스템
- 보안
  - 원하지 않는 행위나 영향으로부터 정해진 대상을 안전하게 유지하는 것
  - 보안의 대상 : 재화, 문서, 사람과 같이 물리적인 형체
  - 보안 방법 : 경비원, 금고, 자물쇠 등과 같은 물리적인 격리나 접근 제한

보안개념의 변화

- 컴퓨터 보안
  - 대상이 단순한 물리적인 객체일 뿐만 아니라 정보와 같은 추상적인 객체도 포함됨
  - 유/무선 통신으로 인해 정보의 물리적인 보호가 어려워짐.
    - 추상적인 접근 제약이 필요함 (예: 암호화)
- 정보보안
  - 물리 대상뿐만 아니라 추상적인 대상인 정보까지도 접근 제약을 가능하게 하는 행동과 시스템에 관련된 분야

보안 모델 / 수준

- 무보안
  - 보호를 전혀 고려하지 않음
- 은닉 차원의 보안
  - 정보가 존재하지 않는 것처럼 꾸밈
- 지역보안
  - 물리적으로 대상을 방어함,
  - 대상은 잘 이동하지 않는 것으로 한정함
- 네트워크 보안
  - 정보가 공공상의 또는 안전하지 않은 네트워크를 통해 이동하도록 허용함

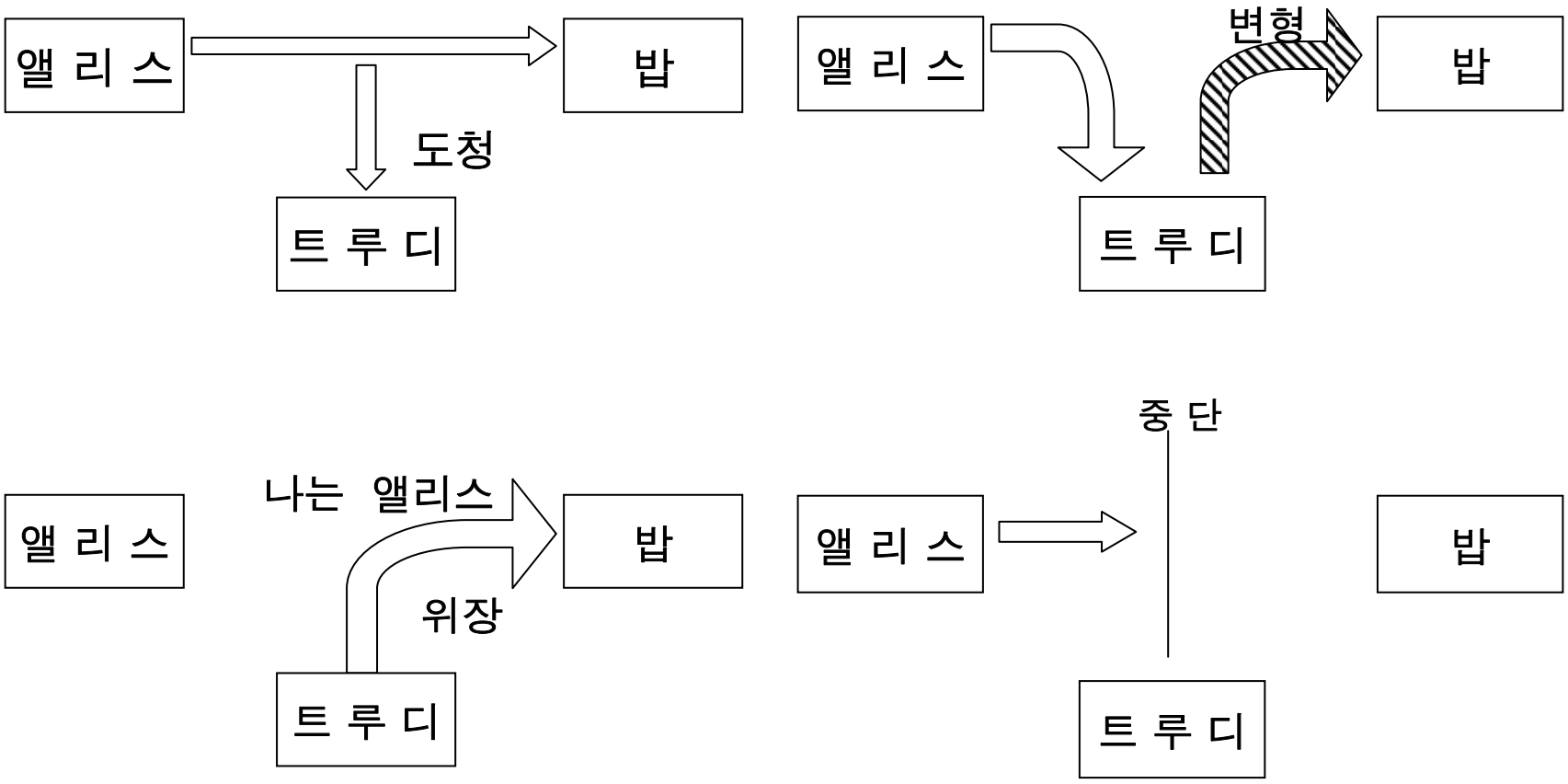
보안 원칙

- 비밀성 (confidentiality)
  - 정보를 제 3자가 볼 수 없도록 하는 기능
- 무결성 (integrity)
  - 정보가 변경되지 않게 하는 기능, 또는
  - 정보가 변경됨을 감지할 수 있는 기능
- 인증 (authentication)
  - 정보 생성자의 신원을 확인할 수 있는 기능
- 가용성 (availability)
  - 정보를 원할 때 항상 사용할 수 있는 기능

보안에 대한 공격

- 도청 (interception, eavesdrop)
  - 공격자가 정보를 얻는 행위
- 위장 (fabrication)
  - 공격자가 합법적인 사용자로 가장하는 행위
- 변형 (modification)
  - 공격자가 정보를 변경하는 행위
- 중단 (interruption)
  - 정보가 받을 사람에게 도달하지 못하게 하는 행위

공격의 분류



공격의 적극성

- 소극적 공격
  - 정보 자체에 변형을 가하지 않는 공격
  - 정보를 엿듣거나 감추어진 내용을 찾아내는 행위
  - 도청이 이에 해당됨
- 적극적 공격
  - 정보를 변경하거나, 전송되지 못하도록 방해하거나, 가짜 정보를 만드는 등의 행위
  - 위장, 변형, 중단이 이에 해당됨

악성코드

- 공격의 구체적인 형태
- 바이러스 (virus)
  - 다른 프로그램에 자신의 코드로 감염시키는 코드. 감염된 프로그램은 감염을 확산시킴
- 웜 (worm)
  - 자신을 계속해서 복제하는 프로그램, 인터넷을 통해서도 자신을 복제함
- 트로이목마 (Trojan horse)
  - 무해해 보이는 프로그램 내부에 유해한 프로그램 코드가 들어 있음

OpenSSL

- Young과 Hudson이 개발한 SSLeay 라이브러리 확장
- TLS / SSL 지원 라이브러리 패키지
- 암호 알고리즘 지원
- 커멘드라인 쉘 프로그램
- OpenSSL 응용프로그램: Apache-SSL, GLOBUS

Unix 시스템에서 OpenSSL의 설치

- http://www.openssl.org/source/에서 최신 버전 다운로드

```
# gunzip openssl-xxx.tar.gz
# tar xvf openssl-xxx.tar.gz
# cd openssl-xxx
# ./config
# make
# make test
# su
# make install
```

OpenSSL command line 실행

```
# openssl
OpenSSL> genrsa –out Apriv.pem
OpenSSL> quit
# openssl genrsa –out Apriv.pem
```

OpenSSL 제공 library 사용

```
# gcc –g –Wall –o testRSA testRSA.c –
  lcrypto
```

프로젝트 #1

- P1a
  - 난이도 : 중간 (소켓 프로그래밍의 경험이 있으면 쉽게 할 수 있다.)
  - 내용 : 2개의 콘솔 창에서 수행되는 채팅 프로그램을 구현하시오.
- P1b
  - 난이도 : 높음 (OpenSSL 매뉴얼을 참조하여 프로그래밍 해야 함)
  - 내용 : P1a 프로젝트는 소켓 프로그래밍으로 구현되어 있다. P1b 프로젝트는 이를 OpenSSL에서 제공하는 BIO 라이브러리를 사용하여 구현하는 것이다.
- P1c
  - 난이도 : 매우 높음
  - 내용 : P1b의 내용에 파일을 전송하는 기능을 추가하시오.

프로젝트 P1a (요구조건)

- 1 대 1 채팅만을 지원함.
- 시작하면 각 사용자는 2개의 프로세스(Sender, Receiver)를 실행하며 각 프로세스는 각 콘솔 창에서 수행됨. 즉, 한 명의 사용자당 2개의 콘솔 창을 사용함.
- Sender는 사용자의 키 입력을 읽고 엔터키가 입력되면 이를 상대방 Receiver에게 보내는 역할을 함.
- Receiver는 상대편 Sender가 보낸 문자열을 받아서 화면에 출력함.
- Sender는 시작할 때 상대방 ip 주소와 port 번호를 파라미터로 입력, ip 주소와 port 번호를 입력하지 않았을 때는 default 값으로 수행됨
- Receiver는 시작할 때 port 번호를 파라미터로 입력, port 번호를 입력하지 않았을 때는 default 값으로 수행됨.
- Sender에서 quit 문자를 입력하면 그 문자를 상대방 Receiver에게 보내고 자신은 종료하도록 하시오.
- Receiver는 상대방으로부터 quit 문자를 받으면 종료하도록 하시오.

프로젝트 P1c (추가 요구조건)

- Sender에 입력할 때 send (또는 한글로 보낸다) 라는 명령어로 시작하면 파일을 전송하도록 하시오. 파일을 전송하기 전에 상대방이 받을지 여부를 묻도록 하시오.
- Receiver는 상대방이 파일을 보낸다는 의사를 보내왔을 때 그 파일을 받을지를 판단하는 메시지를 출력하시오.
- 상대방의 Sender가 yes (또는 한글로 예)를 타이핑하면 파일이 전송됨. 그 파일은 Receiver 실행파일이 있는 디렉터리에 저장되도록 하시오.
- 상대방의 Sender가 yes (또는 예) 이외의 글자를 타이핑하면 전송을 하지 않도록 하시오.