

암호화 채팅

7주차 강의 (2018.10.22)

정보보호 연구실

김근영

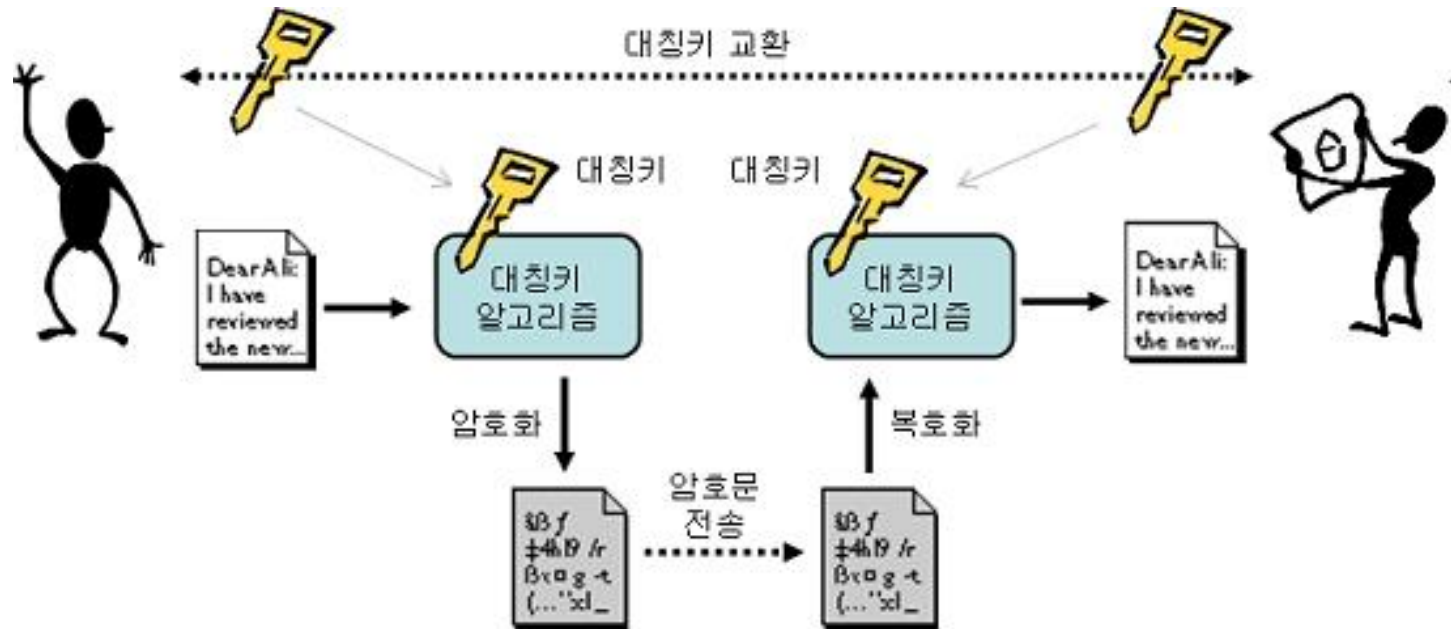
gykim@gmail.com



대칭키 대칭키

■ 대칭키 암호/복호화 방식

- ▶ 암호화에 사용되는 키와 복호화에 사용되는 키가 같음



■ 1:1 통신 프로그래밍

▶ 사용 언어 : C

▶ 목적 : 두 개의 클라이언트가 각자의 메시지를 주고 받는 것이 가능

▶ 조건 :

- 클라이언트 간의 대화 내용을 중간에 확인 가능한 형태가 되어야 함

: 예를 들어 소켓 통신의 경우,

1) 서버가 채팅 서버 생성

2) 클라이언트 A가 서버에 접속, 클라이언트 B가 서버에 접속

3) A가 B로 보내는 메시지는 서버를 경유하므로, 서버에서는 메시지 출력 가능

- client.h client.c 및 server.c 파일 제공

실습 과제

1:1 통신

Compile시 -pthread 옵션

■ 예시 화면

▶ 서버

- 각 클라이언트 메시지 중계

▶ 클라이언트 2

- bbb

- 평문 메시지 전송

▶ 클라이언트 1

- aaa

- 평문 메시지 전송

```
1 Ubuntu 공개서버 x +
cillic@cillic-virtual-machine:~/securityClass/5/noServer$ ./server 12346
[!] New User : 127.0.0.1
[!] New User : 127.0.0.1
[bbbb] Hello
[aaaa] Hi
[bbbb] Nice to meet you
[aaaa] Me too
█

1 Ubuntu 공개서버 x +
cillic@cillic-virtual-machine:~/securityClass/5/noServer$ ./client 127.0.0.1 12346 aaaa
[bbbb] Hello
[aaaa] Hi
[bbbb] Nice to meet you
[aaaa] Me too
█

1 Ubuntu 공개서버 x +
cillic@cillic-virtual-machine:~/securityClass/5/noServer$ ./client 127.0.0.1 12346 bbbb
[bbbb] Hello
[aaaa] Hi
[bbbb] Nice to meet you
[aaaa] Me too
█
```

1:1 암호화 통신

■ 1:1 암호화 통신 프로그래밍

▶ 사용 언어 : C

▶ 목적 : 두 개의 클라이언트가 각자의 메시지를 암호화하여 주고 받는 것이 가능

▶ 조건 :

- 클라이언트 간의 대화 내용을 중간에 확인 가능한 형태가 되어야 함

- : 예를 들어 소켓 통신의 경우,

- 1) 서버가 채팅 서버 생성

- 2) 클라이언트 A가 서버에 접속, 클라이언트 B가 서버에 접속

- 3) A가 B로 보내는 메시지는 서버를 경유하므로, 서버에서는 메시지 출력 가능

- 서버는 암호화된 메시지, 두 클라이언트는 정상 메시지

- 암호화 및 복호화는 클라이언트에서 진행

- 해당 과제의 암호화 및 복호화에는 블록 암호 사용

- 이전 과제의 암호화 및 복호화를 참고

1:1 암호화 통신

■ 예시 화면

▶ 서버

- 각 클라이언트 메시지 중계
- 단, 암호화되어 있어 가독 불가

▶ 클라이언트 2

- bbb : qwe
- 평문 -> 암호화하여 송신
- 암호문 -> 복호화하여 수신

▶ 클라이언트 I

- aaa : qwe
- 평문 -> 암호화하여 송신
- 암호문 -> 복호화하여 수신

```
1 Ubuntu 공개서버 x +
cillic@cillic-virtual-machine:~/securityClass/5$ ./server 55555
[!] New User : 127.0.0.1
[!] New User : 127.0.0.1
[bbb] E
      wd²;c~:¡¢![]=ºNpZIIP      ƒŷ[] ŷ"F¥º[R# # 6XºH=|  s²I>³[]T>j¶[aaa]
Qñ!?V«c 15'UnG7  ¥¢#q      []H d-np_i&[] :`CD祥$/
XshellXshellTnp
      P P[aaa]  ь2z[]&8 x¹[]9'¼-n[] bb] >_
4¾mK []G]i&
³[nbH;ÿyººpé¾;Df[2VI(FHl n¾«qB[]0°†¢4¹

1 Ubuntu 공개서버 x +
lXshellXshellXshellXshellXshellXshellXshellXshellXshellXshellXshellXshellXshell
cillic@cillic-virtual-machine:~/securityClass/5$ ^C
cillic@cillic-virtual-machine:~/securityClass/5$ ^C
cillic@cillic-virtual-machine:~/securityClass/5$ ^C
cillic@cillic-virtual-machine:~/securityClass/5$ ./client 127.0.0.1 55555 aaa
>> Input Key : qwe
[bbb] Hello
[aaa] Hi
[aaa] Nice to Meet you
[bbb] Me too
[]

1 Ubuntu 공개서버 x +
Connection established.
To escape to local shell, press 'Ctrl+Alt+]'.

Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.15.0-36-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

129 packages can be updated.
0 updates are security updates.

New release '18.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Mon Oct 22 17:32:52 2018 from 168.188.129.178
cillic@cillic-virtual-machine:~$ ^C
cillic@cillic-virtual-machine:~$ ^C
cillic@cillic-virtual-machine:~$ cd securityClass/5/
cillic@cillic-virtual-machine:~/securityClass/5$ ./client 127.0.0.1 55555 bbb
>> Input Key : qwe
[bbb] Hello
[aaa] Hi
[aaa] Nice to Meet you
[bbb] Me too
```

제출 요령

■ 보고서 (*.pdf)

- ▶ 문서는 PDF로 변환하여 제출
- ▶ 과제 해결 과정
 - 과제를 어떻게 이해했는지
 - 어떻게 해결했는지

■ 소스코드 (*.c / *.h / Makefile)

- ▶ 과제 해결에 작성한 코드

❖ 소스코드

- 1) 실습 부분 코드
 - 코드 분석
 - 1:1 암호화 통신

제출 요령

■ 제출 방법

- ▶ 사이버 캠퍼스 (e-learn.cnu.ac.kr)
- ▶ 강의 > 과제제출
 - 사이버 캠퍼스로 제출한 과제만 인정

■ 제출 기한

- ▶ 2018 10. 23 00:00:00 ~ 2018 10. 29 23:59:59
 - 추가 제출 기한 : 2018 10. 30 ~ 2018 11.5 23:59:59