

# 16. FTP

grandmarnier

# Contents

- Review
  - Protocol
  - OSI Model
- FTP
  - What is that?
  - characteristics
  - Active vs Passive
  - functions
- SFTP
  - Secure FTP
  - sftp vs ftps
- Practice Session

# 깊고 넘어가기

- Protocol
  - 표준화된 통신규약
  - 통신을 원하는 두 개체(client) 사이에서
    - 무엇을 (electronic signal(0/1), (hyper)text, files... )
    - 어떻게 (format: meta data, encoding, encrypted...)
    - 언제 (periodic, continuous, interactive, non-interactive...)

통신할 것인가를 서로 정하는 약속

# 짚고 넘어가기

- OSI Reference Model

- 앞으로의 설명에 필요한 계층들

OSI	TCP/IP
Layer 7 Application	Application Telnet, FTP, NFS, NIS
Layer 6 Presentation	Session e.g. RPC
Layer 5 Session	Transport Sockets/Streams - TLI
Layer 4 Transport	TCP      UDP
Layer 3 Network	Network IP + ARP/RARP/ICMP
Layer 2 Data Link	Physical Protocol Ethernet/TR/FDDI/PPP
Layer 1 Physical	Transmission medium Coax, Fiber, 10baseT..

네트워크 계층(Network layer)

- Data delivery type

전송 계층(Transport layer)

- TCP : data integrity guaranteed
- UDP: fast, but risky sometimes

표현 계층(Presentation layer)

- encoding
- encryption & decryption

응용 계층(Application layer)

- service type

20%

# Status

## ✓ Review

— Protocol

— ~~OSI Reference Model~~

## • FTP

- What is that?
- characteristics
- Active vs Passive
- functions

## • SFTP

- Secure FTP
- sftp vs ftps

## • Practice Session

# FTP

- What is that? File Transfer Protocol
  - 파일 전송(대용량일수록)에 특화되어 있음
  - TCP/IP 기반의 응용 계층 프로토콜
  - RFC에 규격이 명시되어 있음
  - 1950년대 인터넷의 초기 모델인 ARPANET에서부터 고안되었으며 1970년대에 TCP/IP가 만들어진 후 이 프로토콜 기반으로 새로 개발됨

# FTP

- Characteristics

- 형식이 간단하며 유연함

Format : 

MAC header	IP header	TCP header	data
------------	-----------	------------	------

- 서버와 클라이언트는 둘 다 2개 이상의 포트를 사용

- Port 21 : 접속용 포트
- Port 20 or 1024~ : data 전송용 포트

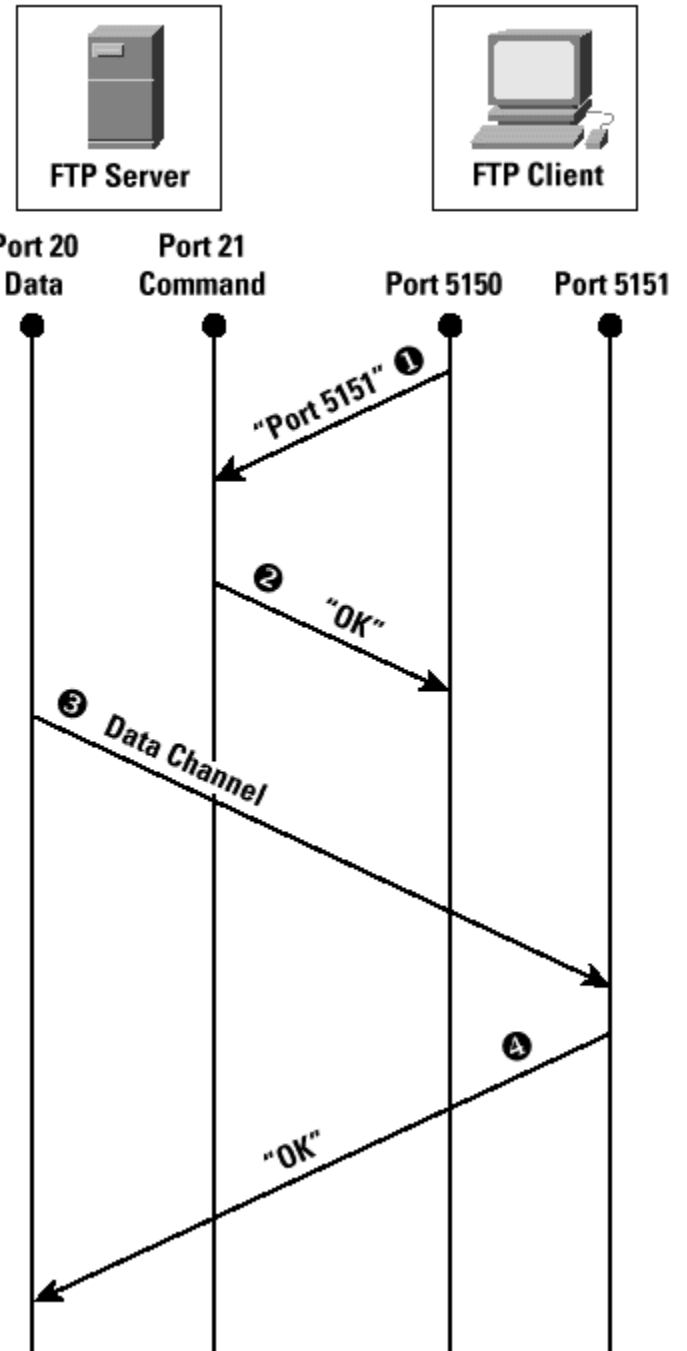
- 데이터 전송 중간에 종료, 이어받기가 가능

# FTP

- Active vs Passive
  - Active mode

FTP 나 WS-FTP, CuteFTP 같은 FTP 클라이언트 프로그램에서 사용  
서버에서는 전송용으로 port 20을 사용

1. Port 21 에 클라이언트가 접속하여 자신이 data 전송용으로 사용할 포트를 알려줌
2. 서버는 ack로 응답
3. port 20에서 클라이언트가 지정한 포트로 접속

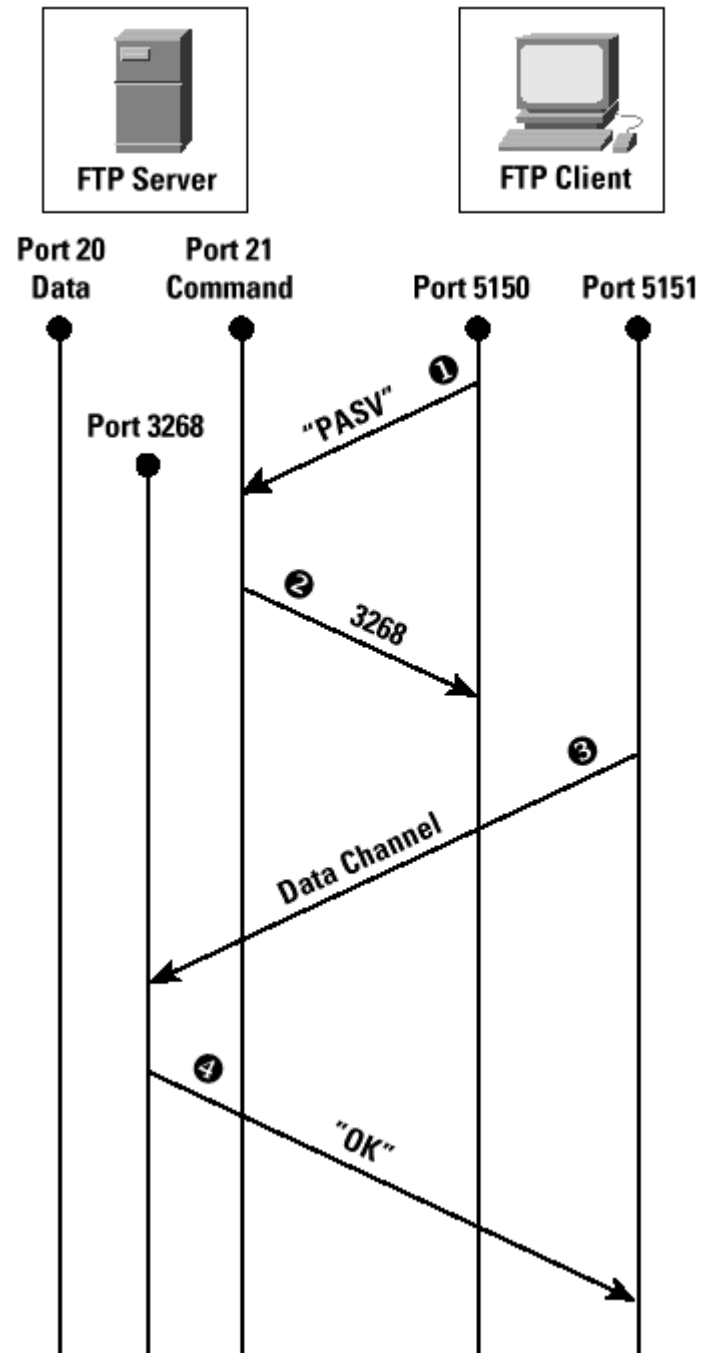


# FTP

- Active vs Passive
  - Passive mode

보통 웹 브라우저에서 사용  
서버는 1024번 이상의 포트 사용

1. Port 21 에 클라이언트가 접속하여 PASV 신호를 보냄
2. 서버가 클라이언트에게 전송용으로 사용할 포트를 알려줌
3. 클라이언트는 다른 포트를 열어 서버가 알려준 포트에 접속



# FTP

- Active vs Passive

- Active

- 서버가 클라이언트로 접속하는 과정이 필요함  
=> 클라이언트가 방화벽을 사용하거나,  
ftp 를 이해하지 못하는 공유기를 지나는 경우  
ftp 접속에 장애가 생김
- 서버가 방화벽을 쓰는 경우에 적합

- Passive

- 서버도 1024~65536 사이의 모든 포트를 열어두어야 함
- 보통 ftp 데몬은 사용하는 포트를 제한할 수 있는 기능을 지원
- 클라이언트가 방화벽을 사용하는 경우에 적합

# FTP

- functions
  - 사용자 관리
    - : 익명 사용자, 접속 가능 시간 제한
  - 파일 관리
    - : 전송 가능한 파일 종류 선택
  - 권한 관리
    - : 접근 가능 디렉토리 정의, 업/다운로드 권한
  - 특정 사용자, IP제한하기
  - 그 외 등등

50%

# Status

## ✓ Review

- Protocol
- OSI Reference Model

## ✓ FTP

- What is that?
- Active vs Passive
- characteristics
- functions

## • SFTP

- Secure FTP
- sftp vs ftps

## • Practice Session

# SFTP

- Secure FTP
  - FTP의 단점 : ID, passwd가 평문으로 전송됨
  - 이를 보완하기 위해 ssh(secure shell) 위에서 scp를 이용하여 ftp와 비슷한 서비스를 제공하는 것이 sftp
  - 엄격히 말하면 ssh FTP 이 정확함
  - 접속포트는 ssh와 같은 포트인 22번을 사용
  - 따라서 설정도 /etc/ssh\_config, /etc/sshd\_config에 따름

# SFTP

- SFTP vs FTPS
  - ftps : FTP over a SSL
  - ssl : secure socket layer
  - ftp와 동일하며 단지 SSL을 이용한다는 차이  
ex) 21번 포트 사용
  - 보안이 우수하며 ftp 의 장점을 그대로 이어받음

# SFTP

- SFTP vs FTPS

- sftp의 장점

1. ssh 위에서 사용되므로 보안이 좋음
2. ssh 와 설정을 공유하므로 간편함

- sftp의 단점

1. ftp보다 복잡하므로 속도면에서 불리
  2. ftp와 달리 클라이언트의 디렉토리 이용을 제한할 수 없으므로 해킹의 가능성이 있음
- => 요즘은 복잡하지만 제어 가능

65%

# Status

## ✓ Review

- Protocol
- OSI Reference Model

## ✓ FTP

- What is that?
- characteristics
- Active vs Passive
- functions

## ✓ SFTP

- Secure FTP
- sftp vs ftps

## • Practice Session

# Practice Session

- utilities
  - Windows
    - Windows 탐색기
    - WinSCP
    - ✓ filezilla FTP server/client
  - Linux
    - proftpd
    - gssftpd
    - ✓ vsftpd : sparcs ftp 에서 사용

# Practice Session

- FTP 로 접속해보기
- FTP 명령어 사용해보기
- sftp(ssh) 설치 및 설정하기
- vsftpd 설치 및 설정하기
  - option : Filezilla 사용하기

# Practice Session

- FTP 사이트에 접속해보기
  1. <http://ftp.kaist.ac.kr/> ( 거울 )
  2. Filezilla 로 접속하는 법은 이미 했으니 remind만 합시다!
  3. Windows 탐색기에서 <ftp://sparcs.org> 사용해보기

# Practice Session

- FTP 명령어 사용해보기

Putty 에서

1. ftp server(:port)

Ex) ftp bit.sparcs.org

2. Password 입력

3. 다음 명령어들을 이용해보자

ls, cd, pwd, mkdir, rmdir, chmod 등은 셸과 동일

close, disconnect : 연결을 끊음. ftp 세션은 유지

bye, exit, quit : 연결을 끊으며 ftp 도 종료

open server(:port) : 연결 시도

**put, send [filename] : 현재 접속한 곳으로 파일을 보냄**

**get [filename] : 현재 접속한 서버에서 파일을 가져옴**

!caution : 여긴 bash 가 아니므로 tab 자동 완성, wild card, pipe 등은 안 먹힙니다.

# Practice Session

- sftp 설치 및 설정
  - 설치하기
    - `#apt-get install openssh-server`
    - 22번 포트 열어주기
    - ssh 를 설치하면 sftp는 기본적으로 실행 가능함
    - 또한 설정 역시 ssh를 따름
  - 관리하기
    - `/etc/sshd_config`
    - 변경 후 `/etc/init.d/ssh restart` 로 변경사항 적용
    - `Subsystem sftp /usr/lib/openssh/sftp-server (ssh_config)`  
=> sftp 사용 여부, 사용하지 않을 시 #으로 주석처리

# Practice Session

- sftp Options
  - AllowGroups
    - ssh 로그인을 해당 그룹으로 제한한다. 각각의 그룹명은 공백으로 구분한다. 와일드 카드(\* 와 ?)를 사용할수 있다.
  - AllowUsers
    - ssh 로그인을 해당 유저로 제한한다. 사용법은 AllowGroups과 같다.
  - DenyGroups
    - AllowGroups의 반대 역할을 한다. 지정된 그룹은 로그인이 거부된다.
  - DenyUsers
    - AllowUsers의 반대 역할을 한다. 지정된 사용자는 로그인이 거부된다

# Practice Session

- Vsftpd 설치 및 설정

- 설치하기

- : 패키지 관리자 이용

- ```
#apt-get install vsftpd
```

※ client 는 gftp or ncftp 를 사용하자

- 설정파일 : /etc/vsftp.conf

- 변경 후에는 /etc/init.d/vsftpd restart 로 재시작하자

# Practice Session

- Vsftpd Options
  - pasv\_enable=YES
  - pasv\_min\_port= 10001
  - pasv\_max\_port= 10050
  - 패시브 모드 허용 여부
  - 허용 시 포트의 하한과 상한을 정함
  
  - connect\_from\_port\_20=YES
  - 20번 포트의 데이터전송 연결 허용 여부
  
  - session\_support=YES
  - wtmp에 ftp 접속관련 기록을 남길지 결정
  - last 명령어로 ftp접속기록을 확인할 수 있음
  
  - max\_clients= N
  - max\_per\_ip=M
  - max\_clients는 ftp에 접속 가능한 최대 인원
  - max\_per\_ip는 한 IP에서 접속 가능한 최대 인원

# Practice Session

- 익명 연결
  - anonymous\_enable=YES
  - 익명 접속의 허용 여부
  
  - anon\_upload\_enable=YES
  - 익명 사용자에게 파일 업로드를 허용할지의 여부
  - 가능한 익명계정으로 접속한 사용자에게는 업로드 권한을 허용하지 않는 것이 보안에 훨씬 좋음
  
  - anon\_mkdir\_write\_enable=YES
  - 익명 사용자에게 디렉토리 생성권한을 허용할지의 여부
  - 익명계정으로 접속한 사용자에게는 디렉토리 생성권한을 허용하지 않는 것이 보안에 훨씬 좋음

익명 접속은 name : anonymous, password : [이메일 주소] 로  
기본 디렉토리는 ftp의 홈 디렉토리(/etc/passwd 에서 확인)

```
#chown ftp ftp/ , #chmod 744 ftp/
```

로 익명 사용자가 업로드 하고 다른 사람들이 다운로드 받을 수 있게 함

# Practice Session

- listen=YES
- listen\_port=21
- 만약 vsftpd를 standalone으로 서비스하려면 위의 listen지시자를 YES로 설정하고 listen\_port에 서비스할 포트번호(기본 21번)를 지정
  
- local\_enable=YES
- 로컬 계정 사용자들의 접속을 허용할 것인가의 여부를 결정
  
- write\_enable=YES
- ftp전용명령어 중에 write명령어를 허용할 것인가를 결정
  
- pam\_service\_name=vsftpd
- vsftpd에서 PAM설정파일명으로 사용할 파일명을 지정
  
- local\_umask=022
- 로컬계정 사용자들의 umask값을 설정
  
- ascii\_upload\_enable=YES
- ascii\_download\_enable=YES
- ASCII모드 전송의 허용 여부

# Practice Session

- `chroot_list_enable=YES`
- `chroot_list_file=/etc/vsftpd.chroot_list`
- 특정 사용자들에 대하여 자신의 홈디렉토리를 루트디렉토리
- 이를 적용시킬 사용자를 `chroot_list_file`에 등록함
  
- `chroot_local_user=YES`
- 특정 사용자가 아닌 전체 사용자를 대상으로 `chroot()`기능을 적용하여 자기 자신의 홈디렉토리 상위 디렉토리로 이동하지 못하게 함
- "`chroot_list_enable=YES`"와 "`chroot_local_user=YES`" 설정이 모두 YES로 되어 있다면 `/etc/vsftpd.chroot_list`에 등록된 사용자들만 `chroot()`적용을 받지 않음
  
- `ls_recurse_enable=YES`
- ftp 접속에는 ls사용시 `-R` 옵션 허용 여부
- 서버부하등의 이유로 ftp에서 기본적으로는 지원하지 않지만 vsftpd에서는 이 옵션을 지원

# Practice Session

- `idle_session_timeout=600`
- ftp 연결에서 idle타임에 대한 타임아웃값을 설정
  
- `data_connection_timeout=120`
- 데이터 전송시 적용되는 타임아웃값을 설정
- 전송 중간에 이 설정에 의해 전송이 중단될 수 있음
  
- `anon_max_rate=0`
- `local_max_rate=0`
- `trans_chunk_size=0`
- ftp 서비스의 전송속도를 제한
- 초당 byte수를 지정하여 0은 무제한
- 이 설정은 vsftpd가 독립데몬(standalone)모드로 서비스될 때에만 적용
  
- `deny_email_enable=YES`
- `banned_email_file=/etc/vsftpd.banned_email`
- 익명접속시에 기본적으로 사용되는 계정명은 `anonymous`, 패스워드는 email형식
- 이때 패스워드로 사용하지 못하도록 할 email 주소를 지정

# Practice Session

- ftpd\_banner=Welcome to blah FTP service.
- ftp서버로 접속할 때에 안내메세지등을 출력
  
- dirmessage\_enable=YES
- ftp 접속한 사용자가 특정 디렉토리로 이동하였을 때 개별 디렉토리의 메시지를 보여주도록 허용할 것인가(YES) 허용하지 않을 것인가(NO)를 설정
- message\_file=.message
- ftp 접속후에 특정 디렉토리로 이동할 때에 디렉토리 안내메세지 파일로 사용할 파일명을 지정한 것입니다. "dirmessage\_enable" 이 YES로 설정되어 있을 때 적용
  
- ferlog\_file=/var/log/vsftpd.log
- ftp 로그파일의 위치를 결정
  
- xferlog\_std\_format=YES
- 로그파일에 남길 로그파일의 포맷을 기본포맷으로 남길 것인가(YES) 아닌가(NO)를 설정하는 지시자입니다.
- 이 파일의 포맷보다는 vsftpd 로그포맷이 더 상세한 기록을 포함 (디렉토리생성, 로그인 등)
  
- xferlog\_enable=YES
- ftp 접속후에 파일 업로드와 다운로드에 대한 로그 기록 여부

# Practice Session

- Vsftpd, sftp 실습
  - 익명이 접속 가능한 ftp 서버를 만들고 접속 기록을 last 로 남김
  - 익명에게 파일 업로드 권한을 주고 파일을 업로드, 다운로드 해보세요

# Practice Session

- Filezilla 사용하기

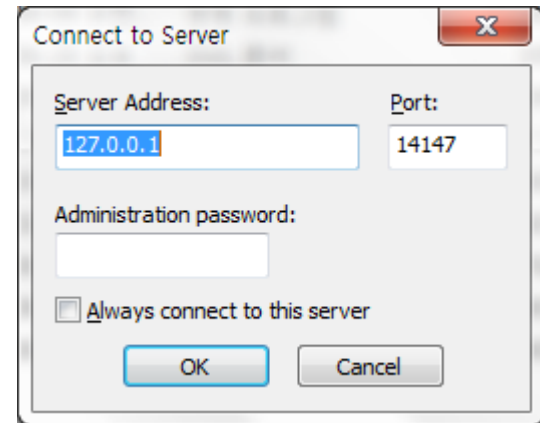
- Server (only in windows)

역시 download 및 설치는 알아서(거의 기본값으로 두시면 됩니다)

1. 설치 후 처음에는 비밀번호가 없으므로 그냥 들어갑니다.

관리자 비밀번호는 나중에 설정가능

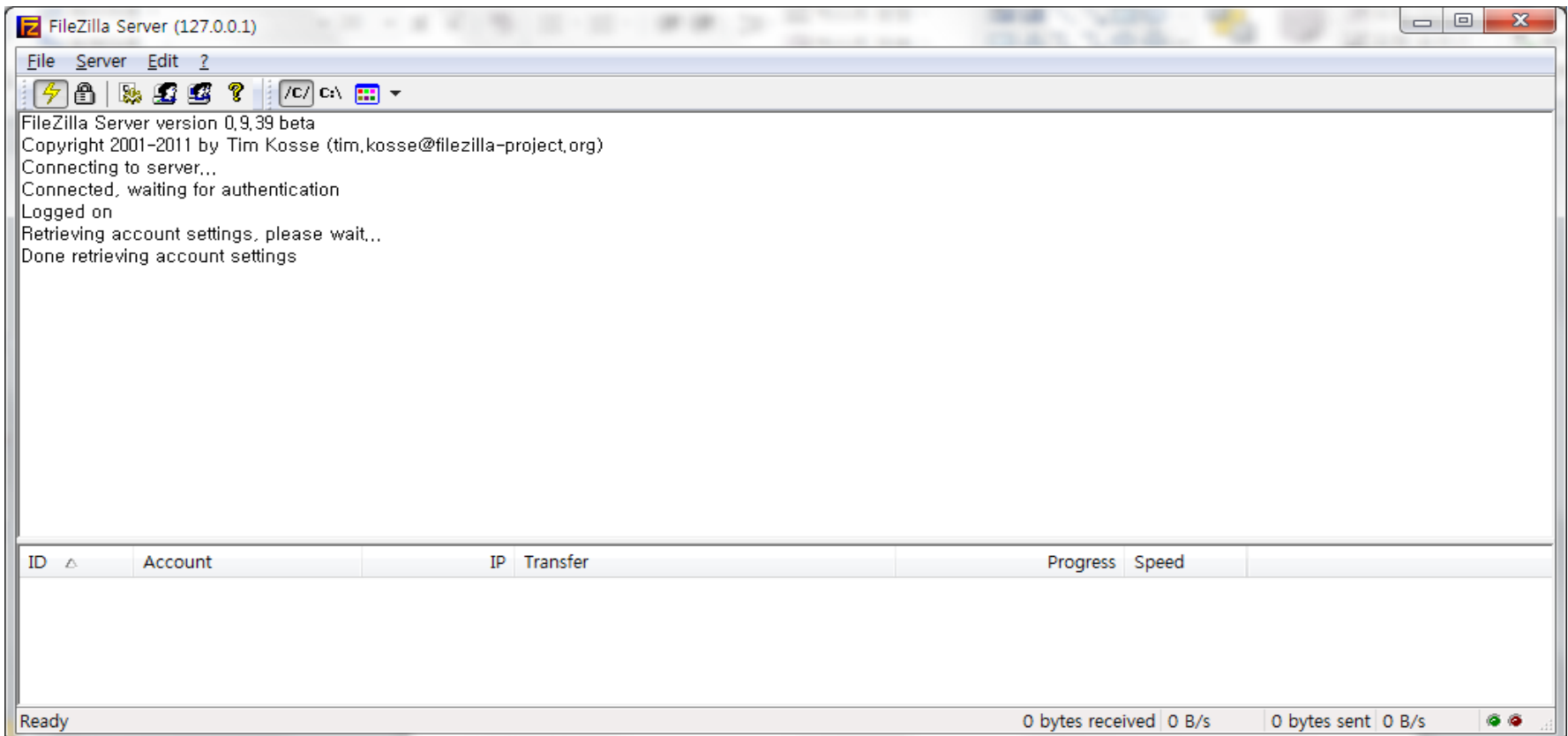
포트의 변경은 상관없음



# Practice Session

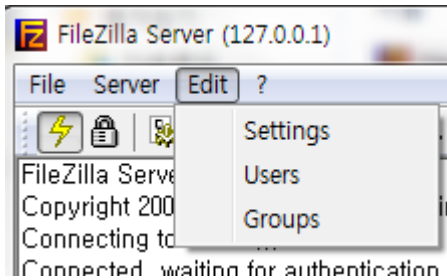
- Filezilla 사용하기
  - Server (only in windows)

성공시 뜨는 창



# Practice Session

- Filezilla 사용하기
    - Server (only in windows)
- edit



- Settings : 서버 설정
- Users : 유저 등록 및 관리
- Groups : 그룹 등록 및 관리

99%

# Status

## ✓ Review

- Protocol
- OSI Reference Model

## ✓ FTP

- What is that?
- characteristics
- Active vs Passive
- functions

## ✓ SFTP

- Secure FTP
- sftp vs ftps

## ✓ Practice Session

# End

## 수고하셨습니다

마지막 1%는 여러분의 몫!

# 출처

1. leeopop의 네트워크 세미나 (2011-02-24)
2. dothack, reniowood의 FTP 훔 세미나 (2010-07-14, 2009-08-06)
3. Zzongaly의 신입생 세미나 (2011-02-21)
4. Computer Networks and Internets , Douglas E. Comer
5. <http://ko.wikipedia.org/>
6. <http://terms.co.kr/>
7. <http://ethdemor.springnote.com/pages/5435733>
8. <http://blog.keun.kr/479>
9. <http://ntfaq.co.kr/3441>
10. <http://daniel.haxx.se/docs/ftp-vs-http.html>
11. <http://www.gpgstudy.com/forum/viewtopic.php?p=118009&sid=7ed377e0189afdc96fc79bc9c9e944ea>
12. [http://touc.tredio.net/bbs/board.php?bo\\_table=faq&wr\\_id=4](http://touc.tredio.net/bbs/board.php?bo_table=faq&wr_id=4)
13. <http://www.seektime.info/entry/FTP-접속설정-FTP-와-SFTP-와의-차이점>
14. <http://www.networksorcery.com/enp/default0403.htm>
15. <http://voals.egloos.com/1670050>
16. <http://blog.keun.kr/entry/VsFTPD-FTP-서버-설치인터넷문서>
17. <http://wiki.filezilla-project.org/Documentation>
18. <http://hyungjae.egloos.com/3810222>
19. [http://community.365managed.com/w\\_service/3778](http://community.365managed.com/w_service/3778)