

# Telnet

---

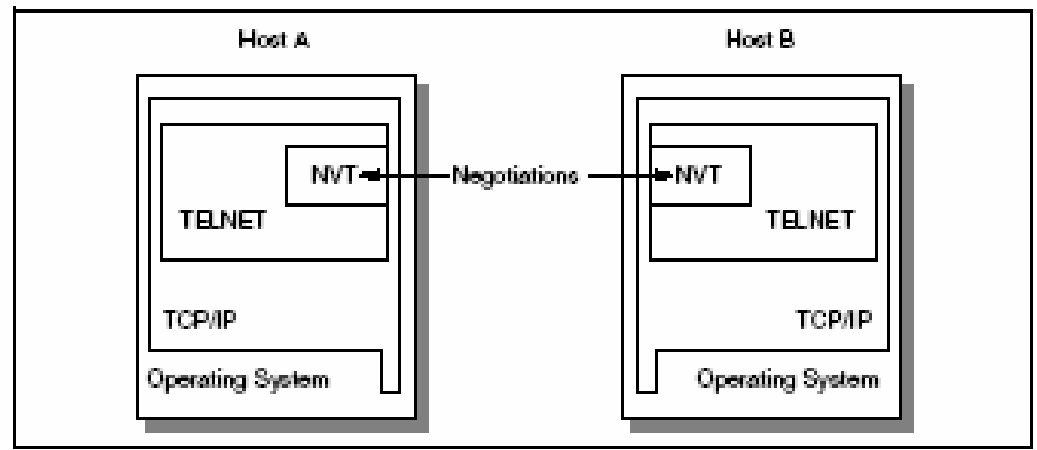
- Protocolo utilizado para la ejecución de procesos en sistemas remotos.
- Emulación de Terminal
- Utiliza las funcionalidades de TCP
- Well Known Service, port number 23



# Telnet

## Operación

- NVT (Network Virtual Terminal)
  - Mapeo de las características de cada terminal a una estructura común
- Arquitectura simétrica
- Negociación de opciones
  - Extiende las funcionalidades no soportadas por NVT



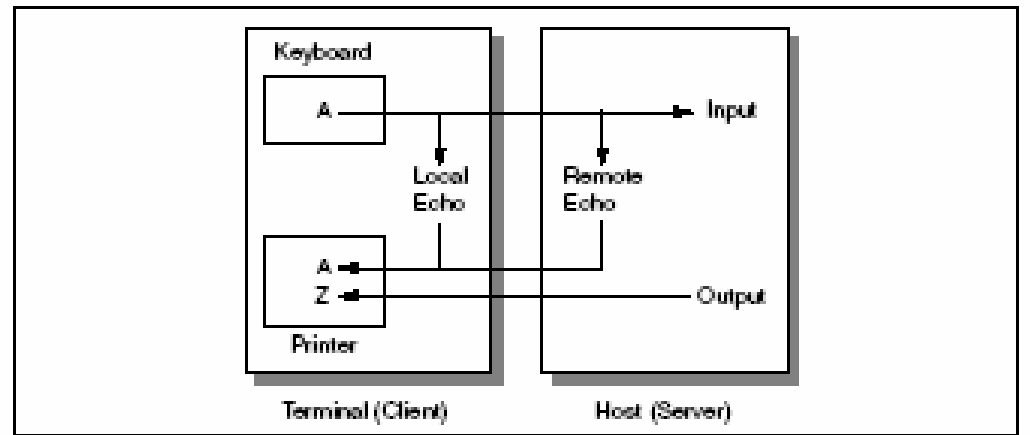
# Telnet

## NVT (Network Virtual Terminal)

NVT define:

Impresora (Printer)

Teclado (Keyboard)



NVT implementa:

Codificación en ASCII de 7 bits

Comunicación en half-duplex

Echo local

# FTP

## File Transfer Protocol

---

Protocolo estandarizado para transferencia de archivos entre equipos remotos

Definido por el RFC 959 y extendido por 2228

Mecanismo de autenticación del cliente ante el servidor



# FTP

## Operación

---

- FTP utiliza TCP para establecer vínculos confiables de extremo a extremo.
- Utiliza 2 Well known services
  - TCP 21      Command port (Control y login)
  - TCP 20      Data port (Transferencia de datos)
    - Dependiendo del modo de transferencia, no siempre la transferencia de datos se realiza sobre el puerto 20 de TCP.
- En ambos extremos se encuentran 2 procesos
  - PI      Protocol Identifier
  - DTP    Data Transfer Process

# FTP

## Operaciones

---

Al usar FTP, el usuario puede realizar alguna o todas estas operaciones:

- Conectarse a un equipo remoto
- Elegir un directorio, listar contenido  
Comandos CD, LS
- Definir un modo de transferencia
- Copiar archivos desde o hacia el equipo remoto
- Desconectarse del equipo remoto

# FTP

## Conexión remota

---

Se utilizan cuatro comandos

**Open** Selecciona el equipo remoto e inicia el procedimiento de login

**User** Identificación del usuario remoto.

**Pass** Autenticación del usuario

**Site** Envío de información al otro host para brindarle servicios específicos.



# FTP

## Transferencia de Archivos

---

Se utilizan los siguientes comandos para copiar archivos del cliente al server o viceversa:

- **Get** Copia un archivo desde el host remoto al host local
- **Mget** Copia múltiples archivos desde el host remoto al host local
- **Put** Copia un archivo desde el host local al host remoto
- **Mput** Copia un múltiples archivos desde el host local al host remoto

# FTP

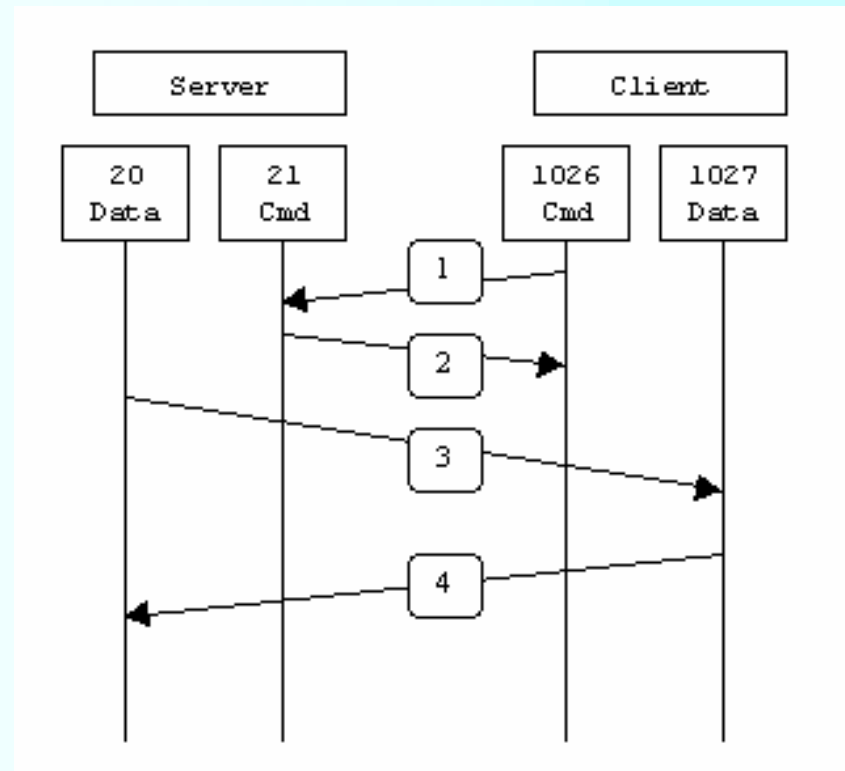
## Modo Activo

El cliente origina una conexión desde un puerto efímero ( $N > 1023$ ) dirigida al puerto de control del servidor FTP (puerto 21).

El servidor acepta la conexión

El cliente escucha en el puerto  $N+1$  y envía el comando `port N+1` al servidor FTP.

El servidor iniciará una conexión contra el puerto indicado por el cliente, utilizando el puerto 20 como origen.



# FTP

## Modo Pasivo

---

- Conocido como FTP de modo pasivo o PASV
- Se definió para resolver el problema que presenta que el server FTP sea quién inicie la conexión utilizada para transferencia de datos
- Se invierte la forma en que se establece el link de datos
  - El cliente es quien inicia ambas conexiones.
  - En lugar de utilizar el comando PORT se utiliza el comando PASV
  - El servidor FTP elige un port number y lo informa al cliente



# FTP

## Anonymous FTP

---

### Anonymous FTP

Se permite la utilización del protocolo FTP a usuarios no identificados.  
Utilizado principalmente para la distribución de archivos públicos

### Terminación de la Sesión FTP

**Quit** Desconecta del host remoto y termina la sesión FTP.

**Close** Desconecta del host remoto, pero deja al cliente FTP corriendo. Se puede utilizar el comando open para iniciar una nueva sesión FTP con otro Host

# FTP activo o pasivo?

---

- ¿Por qué usar modo Activo?
  - Administración simple de firewall en el servidor
    - Solamente se permiten conexiones entrantes al puerto 21
- ¿Por qué usar modo pasivo?
  - No se dejan Well Known ports en estado de listen en el cliente
    - No hay conexiones entrantes en el cliente
  - El administrador del servidor puede especificar el rango de puertos efímeros a ser utilizados
  - Es deseable que un servidor FTP sea por todos los cliente FTP

# TFTP

---

- Protocolo simple de transferencia de archivos
- No es confiable, utiliza UDP en el Well Known Port 69
- Utilizado por dispositivos diskless para obtener el firmware de boot.
- TFTP carece de las funcionalidades de FTP, sólo permite escribir o leer archivos
- TFTP no implementa ningún mecanismo de autenticación

# TFTP

Definido por los RFCs  
783 y 1350

Se definen 2 parámetros:

Filename

Mode

ASCII

Octeto binario

Comando	Descripción
Read Request	Pedido de lectura de archivo
Write Request	Pedido de escritura de archivo
File Data	Transferencia de datos
Data Acknowledge	Confirmación de datos recibidos.
Error	Indicación de error.

# SMTP

## Simple Mail Transfer Protocol

---

Los protocolos básicos de correo de Internet proveen el intercambio de mensajes y notas (mail) entre hosts IP estableciendo conexiones al puerto 25 de TCP.

El intercambio de e-mail se basa en tres protocolos

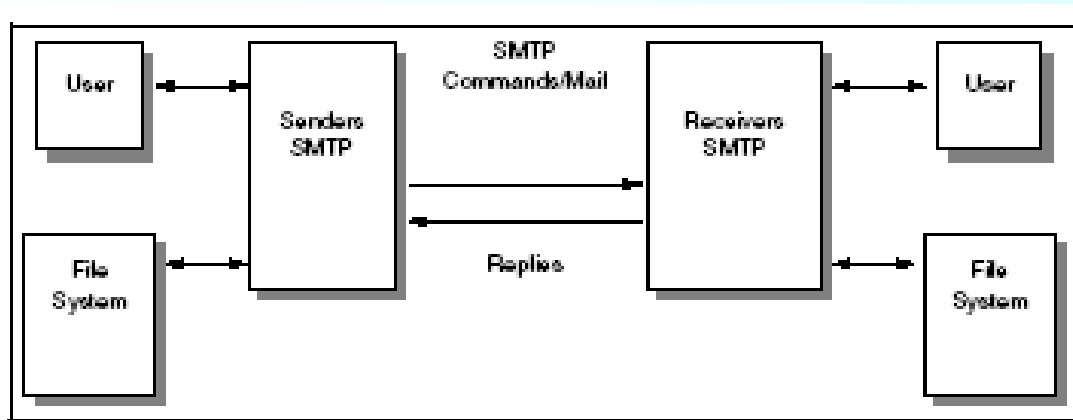
- RFC 821 (Transmission Protocol):
  - Estándar para el intercambio de e-mails entre 2 computadoras conocido como SMTP
- RFC 822 (Message Format):
  - Estándar que define el formato de los mensajes de mail y la sintaxis de los headers. El RFC 1049 describe como encapsular documentos que no sean texto en el body del mail.
- RFC 974 (DNS-MX):
  - Estándar utilizado para el routing de mail usando DNS (Domain Name System)

# SMTP

## Modelo de comunicación

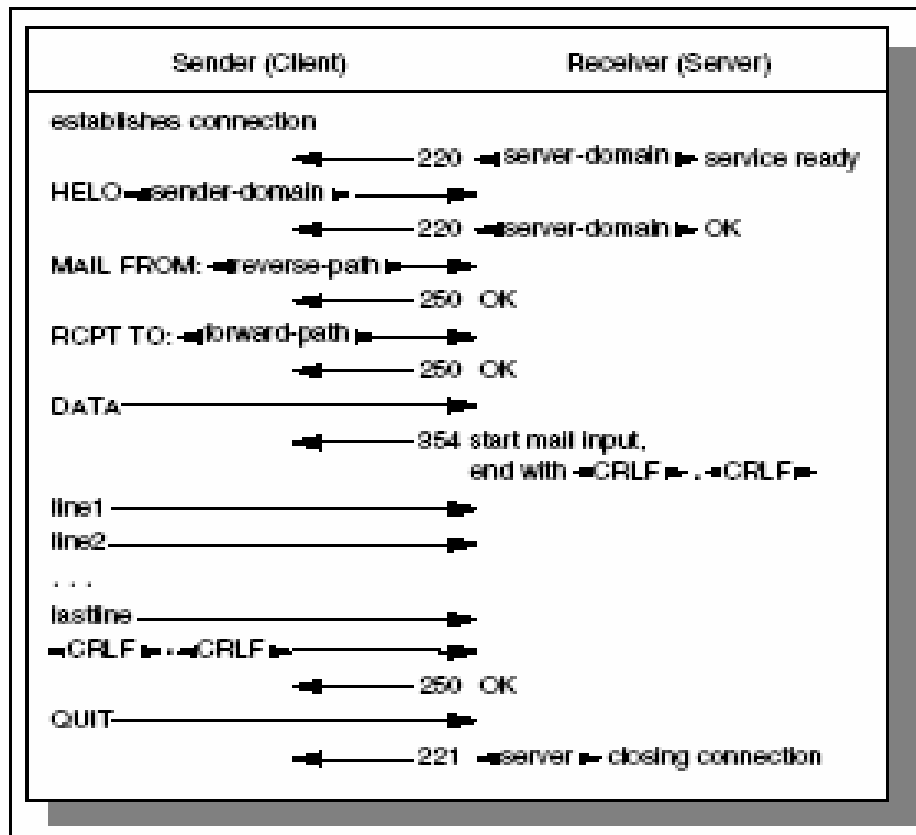
---

- La entidad SMTP que envía los mensajes (cliente) de correo establece una conexión al puerto 25 TCP de la entidad SMTP receptora (servidor)
- El receptor puede ser el destino final o un gateway de SMTP



# SMTP

## Flujo de Datos



- Una vez establecida la conexión, el cliente espera a que el servidor se presente.
- Si el servidor indica que esta listo para recibir e-mails, el cliente especifica la dirección del destinatario y la del remitente, y envía el mensaje.

# SMTP

## Envelopes, Headers and Body

---

El mensaje de e-mail está compuesto por tres partes:

**“Envelope” usado por el MTA para entrega del e-mail.**

MAIL From:<joseperez@fi.uba.ar>

RCPT To:<juanlopez@uyr.com.ar>.

El RFC 821 especifica la sintaxis del “envelope”.

**“Headers” usado por los agentes de usuario MUA.**

Received, Message-Id, From, Date, Reply-To, X-Phone, X-Mailer, To, and Subject.

Cada campo contiene un nombre seguido de “:” seguido del valor

RFC 822 especifica el formato y la interpretación de los campos del “header”. (Headers que comienzan con “X-” son definiciones particulares de cada MUA).

**“body” es el contenido del mensaje.**

RFC 822 especifica que las líneas tienen el formato NVT ASCII (Variación Norteamericana de 7-bit U.S del código ASCII).

Usando el comando DATA, los “headers” se envían primero seguidos de una línea en blanco seguida del “body”

Se denomina “content” al conjunto de “headers” y “body”.

**Redes de Datos – Ing. Marcelo Utard / Ing. Pablo Ronco**

# SMTP

## Comandos

---

El cliente se identifica a si mismo con el comando **HELO**. El valor debe ser el FQDN del cliente.

El comando **MAIL** identifica el origen del mensaje. El siguiente comando, **RCPT**, identifica al receptor del mensaje. Es posible especificar más de un **RCPT** si se requiere enviar el mensaje a más de un destinatario.

El contenido del mensaje de correo es enviado mediante el comando **DATA**. El comando **QUIT**, finaliza el intercambio de e-mails y finaliza la conexión.

El comando **RSET** aborta el envío actual. Cualquier información intercambiada es descartada.

El comando **VRFY** permite verificar la dirección del receptor sin tener que enviar un mensaje.

El comando **NOOP** fuerza a que el servidor responda con el código OK (200).

El comando **TURN** permite intercambiar roles entre el cliente y el servidor. De esta forma se permite el envío de mensajes en sentido inverso sin tener que cerrar la conexión TCP actual e iniciar una nueva.

El RFC 1425 define “Extended SMTP” (ESMTP). Se utiliza el comando **EHLO** en lugar del **HELO**, si **EHLO** es aceptado por el servidor, otras funcionalidad son aceptadas (actualmente siendo definidas)

# SMTP

## Utilización de DNS

---

- Se utilizan los registros MX del DNS para obtener la dirección IP del host destinatario.
- Posibilidad de definir múltiples registros
  - Definición de host preferido y hosts alternativos
  - Balanceo de carga

Ejemplo:

```
fsc5.stn.mlv.fr.  IN MX 0 fsc5.stn.mlv.fr.  
                  IN MX 2 psfred.stn.mlv.fr.  
                  IN MX 4 mvs.stn.mlv.fr.
```

# SMTP

## MIME (Multipurpose Internet Mail Extensions)

---

- SMTP limita a transferir únicamente texto ASCII 7 bits con un máximo línea de 1000 caracteres
- Limitaciones
  - SMTP no puede transferir ejecutables ni archivo binarios
  - No se puede transmitir mensajes de texto en idiomas que utilicen caracteres extendidos del ASCII (ñ, ó, ü, etc).
- MIME es un estándar que soluciona estas limitaciones de SMTP

# MIME

## Operación

---

MIME-Version: 1.0

**MIME-Version** Versión utilizada, debe ser 1.0

**Content-Type** Describe como debe interpretarse el contenido del body. El valor por defecto es: *text/plain; charset=us-ascii*, que indica texto sin formatear de 7-bit ASCII

**Content-Transfer-Encoding** Describe como fue codificado el mensaje en el body

**Content-Description** Una descripción en texto claro del objeto encapsulado en el body. Tiene utilidad cuando el objeto no es leible (datos de audio)

**Content-ID** un valor único que identifica las partes del mensaje.

# POP

## Post Office Protocol

---

- POP3 es el utilizado actualmente
- Se establecen conexiones TCP usando el port 110 en el servidor
- Se entiende que cliente debe descargar todos mensajes de e-mail de un repositorio (Servidor) para poder leerlos offline.
  - Sólo soporta copiado y borrado de mensajes en el servidor
  - Implementa mecanismos de autenticación entre el cliente y el servidor

# IMAP

## Internet Messages Access Protocol

---

- Actualmente se utiliza IMAP4
- Se establecen conexiones TCP usando el port 143 en el servidor
- Diferente a POP3: Se entiende que los mensajes deben residir en el servidor.
  - Los mensajes se administran directamente en el servidor
  - Permite ver encabezados y descargar mensajes por pedido
  - Reestablece sincronismo entre la información en el cliente y la del servidor