

SOCKET RAPPRESENTANO QUINDI DELLE INTERFACCIE NECESSARIE PER GESTIRE LA RETE A LIVELLO APPLICATIVO.

A QUESTO LIVELLO VENGONO UTILIZZATI VARIATI PROTOCOLLI CHE OFFRONO SVARIE FUNZIONALITA', UN ESEMPIO E' IL PROTOCOLLO HTTP.

1/10 SHELL E TERMINALI

SHELL = INTERPRETE DI COMANDI

KERNEL = E' IL NUCLEO E L'UTENTE PER INTERAGIRE INDIRETTAMENTE CON ESSO USA UNA SHELL.

POSSO INTERAGIRE CON LA SHELL TRAMITE TERMINALE.

IL TERMINALE E' UN'EVOUZIONE DELLA TELESCRIVENTE, COMPOSTO DA UNA TASTIERA E UN DISPOSITIVO DI OUTPUT PER VEDERE IL RISULTATO.

SU LINUX CI SONO DUE CONCETTI SIMILI:

- **LE VIRTUAL CONSOLE**, DEI TERMINALI CHE CONDIVIDONO TASTIERA E SCHERMO
- **EMULAZIONE DI TERMINALE**, IMPLEMENTATI TRAMITE PSEUDOTERMINALI, OSSIA FILE CHE FINGONO DI ESSERE UN TERMINALE TRAMITE PROGRAMMA

OGNI PROCESSO SI TROVA DI DEFAULT 3 FILE DESCRIPTOR → NUMERO INTERO CHE RAPPRESENTA UN FILE APERTO PER UN PROCESSO.

0: STANDARD INPUT

1: STANDARD OUTPUT

2: STANDARD ERROR

OGNI PROCESSO HA UNA **CARTELLA DI LAVORO (DIRECTORY)**.

TRA IL TERMINALE E I PROCESSI C'E' LA DISCIPLINA DI LINEA: AD ESEMPIO QUANDO

SCRIVETE UN PROGRAMMA CHE LEGGE UN INPUT, NON VOLETE GESTIRE ANCHE IL

FATTO CHE L'UTENTE CANCELLA UN CARATTERE CON BACKSPACE ECC, E' UN LAVORO DELLA DISCIPLINA DI LINEA.

LA SHELL IN MODALITA' INTERATTIVA, STAMPA IL PROMPT DI COMANDI E QUANDO LEGGE L'INPUT:

- LO SPEZZA IN TOKEN: PAROLE E OPERATORI

• ESPANDERE GLI ALIAS

• FA IL PARSING IN COMANDI SEMPLICI E COMPOSTI

LA SHELL INOLTRE:

- ESEGUE VARIE ESPANSIONI
- ESEGUE LE EVENTUALI REAZIONI DELL'I/O
- ESEGUE IL "COMANDO"

3/10

LA SHELL È UN INTERPRETE DI COMANDI, SPESSA USATA IN MANIERA INTERATTIVA. ALCUNI COMANDI VENGONO ESPANSI;

- $\$VAR$, $\${VAR}$ → LA SHELL ESPANDE QUEL NOME NEL CONTENUTO DELLA VARIABLE.

- LE GRACCE POSSONO ESSERE USATE ANCHE PER INSERIRE DELLE STRINGHE ALL'INTERNO DI ALTRE STRINGHE.

- USANDO LA TILDE (\sim) TI PORTA NELLA HOME DIRECTORY

- SE IO VOGLIO IDENTIFICARE TUTTI I FILE CHE HANNO PER ESEMPIO ESTENSIONE **.C** NELLA DIRECTORY, FACCIO $*.C$; L'ASTERISCO STA PER UN NUMERO ARBITRARIO DI CARATTERI QUALSIASI, QUINDI NELL'ESEMPIO QUALSIASI COSA CHE FINISCA CON **.C**

- (ESPANSIONI) PER FAR CONTI ALLA SHELL

- $\&$ CMD

SE IO VOGLIO CHE LO STANDARD INPUT DI UN COMANDO NON SIA IL TERMINALE MA UN FILE FACCIO $< NOMEFILE$.

SE MENTO $> NOMEFILE$ QUELLO CHE IL PROGRAMMA SCRIVE VERSO LO STANDARD OUTPUT VA A FINIRE DENTO IL FILE.

I COMANDI POSSONO ESSERE:

- **SEMPLICI**: UNA SEQUENZA DI PAROLE SEPARATE DA BLANK, DOVE NON COMPARE LA PIPE

- **PIPELINE**: SEQUENZA DI COMANDI, MESSI "IN PIPE"