

1 L'AMMINISTRAZIONE DI SISTEMA

- CHI È L'AMMINISTRATORE DI SISTEMA?

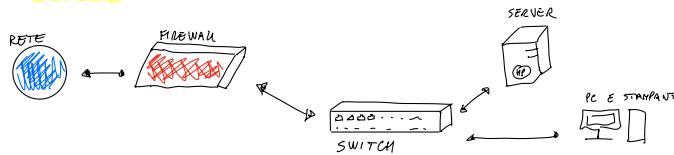
↳ COWI CHE AMMINISTRA (GESTISCE / MANTIENE) IN SISTEMA

- UN SISTEMA PUÒ ESSERE COMPOSTO:

↳ UN SINGOLO PC / DA N PC ED N SERVER / ESSERE UN INSIEME DI ALTRI SISTEMI

- DEF: IN SISTEMA È UN INSIEME DI ELEMENTI (E/O SISTEMI) CHE INTERAGISCONO TRA LORO IN MODO COORDINATO PER POTER RAGGIUNGERE I RISULTATI PER I QUALI IL SISTEMA STESO È STATO DEFINITO

- ESEMPIO: SCHEMA DEL SISTEMA,



- UN SISTEMA SI CAMBIA SOLO QUANDO NON FA PIÙ ANCHE CHE DEVE FARLE,

OPPURE IL NUOVO SISTEMA HA DEI VANTAGGI PALESENTEMENTE SUPERIORI RISPETTO

AL VECCHIO

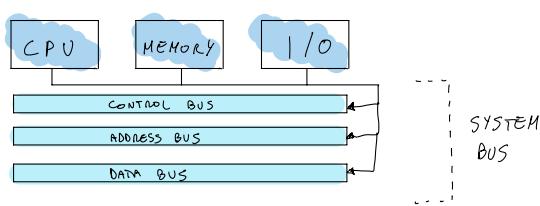
- UN SYS ADMIN DOVREBBE CREARE E MANTENERE LA DOCUMENTAZIONE SUL SISTEMA

↳ MANTENERE PER LEGGE LA DOCUMENTAZIONE SU PUNTI CRITICI (ES. DSS)

↳ DEVONO ESSERE DEI MOTIVI Vaudi PERCHÉ CAMBIARE SISTEMA COSTA

2 HW AND RAID

MACCHINA DI VON NEUMAN



- LA SCHEDA MADRE DI UN COMPUTER È UNA SCHEDA FORMATA DA CHIP, CONDENSATORI, RESISTENZE CONEGATI TRA LORO ATTRAVERSO PISTE DI MATERIALI CONDUTTIVI

↳ CONTIENE LA CPU, LE MEMORIE E GLI SLOT (PIRESE) PER LE SCHEDE DI ESPANSIONE CON LE QUALE POSSIAMO CONEGARE LE PERIPHERICHE

MEMORIE

↳ SUDDIVISIONE: MEMORIE CENTRALI / PRINCIPALI, MEMORIA DI MASSA, MEMORIE ESTERNE (USB,...)

↳ CENTRALI: N° UNITÀ DI INFORMAZIONI. SI DIVIDONO IN:

↳ MEM. DI MASSA, PERCÒ ESTERNE

↳ RAM, ROM (READ-ONLY) E CACHE

↳ MASSA: HANNO LO SCOPO DI CONSERVARE I PROGRAMMI E I DATI IN MODO PERMANENTE

CPU

↳ ESEGUE ISTRUZIONI DI UN PROGRAMMA

↳ MILI DI ISTRUZIONI AL SECONDO

ARCHITETTURA DI UN HOST (SERVER)

↳ RACK NORMALE → A DIFFERENZA DI UN PC, IN UN SERVER SONO RIDONDANTI PIÙ COMPONENTI POSSIBILI (SICUREZZA X I GUASTI)

↳ SERVER TOWER (IN VERTICALE) OPPURE RACK(-ABIN) OSSIA INSERIBILI DENTRO UN ARMADIO RACK

↳ SISTEMA BLADE: COME UN UNICO GRANDE COMPUTER SUDDIVISO IN VARI MINI-COMPUTER (OSSIA IN VARI LANES, OGNIUNO INDIPENDENTE COME SE FOSSE UN UNICO GRANDE PC)

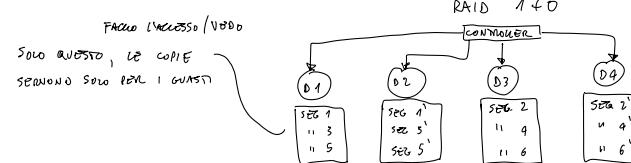
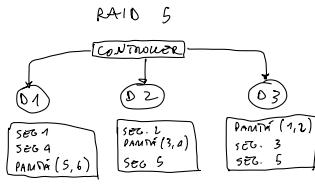
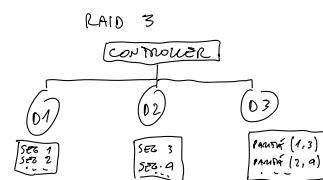
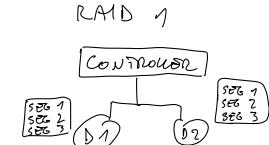
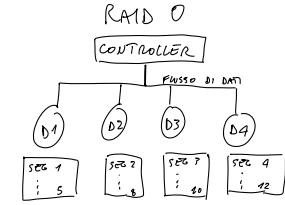
↳ IN GENERALE, È TUTTO UGUALE AD UN CLASSICO PC SOLO CHE È TUTTO PIÙ RIDONDATO E IN MAGGIOR QUANTITÀ

↳ PRESENZA DI ILO (INTEGRATED LIGHTS-OUT): SISTEMA AVANZATO DI AMMINISTRAZIONE REMOTA

↳ FISICALMENTE È UNA PORTA ETHERNET

RAID - REDUNDANT ARRAY OF INEXPENSIVE DISK

- DEF: È UNA ASTRATZIONE CHE, DATO UN GRUPPO DI DISCHI DI SOTTO OLOGENETI TRA LORO, COSTRUISCE UNA UNITÀ DI STORAGE LOGICA, CHE VENNE VISTA DAL SISTEMA OPERATIVO COME UN'UNICA MEMORIA DI MASSA INDIPENDENTEMENTE DAL TIPO DI RAID CHE VIENE USATO E DAI DISCHI CHE LO COMPOSSENO.
 - POSSONO ESSERE UNI OPPURE SW
 - GESTITO DAI SOFT
 - GESTITO DA UN HW PARTICOLARE
 - NEI SERVER, IL RAID, È GESTITO DIRETTAMENTE DAL CONTROLLER DEI DISCHI (RAID HARDWARE)
 - RAID 0** (DETTO ANCHE STRIPPING): SISTEMA IN grado di separare i dati in segmenti e dividerli nei vari dischi
 - ALTE PRESTAZIONI, SCARSSISSIMA SICUREZZA (SI ROMPE 1 DISCO E PERDiamo TUTTI)
 - RAID 1** (o MIRRORING): MANTIENE UNA DUPLICE COPIA DEI DATI IN DUE (O PIÙ) DISCHI
 - MAGGIOR VELOCITÀ IN LETTURA, MINOR VELOCITÀ IN SCRITTURA (SCONTRARE + VOLTE LA STESSA COSA)
 - MASSIMA RIDONDANZA DEI DATI
 - RAID 3**: SIMILE ALLO STRIPPING, SOLO CHE IN AGGIUNTA HA UN DISCO IN CUI VENGONO MERCHEGGIATE LE STRIPPE DI PARITÀ: HAPPA CHE PERMETTE LA RICOSTRUZIONE IN CASO DI GUASTO AD UN DISCO
 - ALMENO 3 DISCHI
 - ALTE PRESTAZIONI IN ACCESSO A FILE GRANDI, MA LETTURE E SCRITTURE NON SONO SIMULTANEE
 - RAID 5**: COME IL RAID 3, SOLO CHE INVECE DI AVERLE TUTTE LE PARITÀ SU UN SOLO DISCO SONO SPARSE IN TUTTI I DISCHI (EFFICIENTE USO DELLO SPAZIO SPARGENDO LE PARITÀ)
 - USATO NEI GROSSI SISTEMI MULTI-USUARI DOVE LE OPERAZIONI DI SCRITTURA SONO MOLTO INFERIORI A THOSE IN LETTURA
 - POTREBBE CONTINUARE A LAVORARE ANCHE PERDENDO, MENEGHÉ RAID 3 SE PERDesse UNO DI PARITÀ BISOGNerebbe PERMETERE ALTRE PRESTAZIONI, FAULT TOLERANCE PARM A 1 (MEDIA PROTETTONE)
 - RAID 6**: COME IL 5, SOLO CHE È PIÙ RIDONDANTE MA INEFFICIENTE QUANDO NON VIENE USATO IN UN N° LIMITATO DI DISCHI
 - PERMETTE UN FAULT TOLERANCE DI 2 DISCHI
 - RAID 1+0**: N° DISCHI ≥ 4
 - VENNE VISTO DAL SISTEMA COME RAID 0, QUANDO "SOTTO" CI SONO 2 × RAID 1
 - NON HA STRIPING DI PARITÀ
 - UTILIZZATO NEI GROSSI SISTEMI DOVE SI NECESSITANO ALTE PRESTAZIONI CON UN OCCHIO ALLA SICUREZZA DEI DATI
- COMPIORTA MIGLIORAMENTI IN: CAPACITÀ (→ "SOMMA" PIÙ DISCHI), SICUREZZA (→ RIPETIZIONE DI DISCHI [MIRRORING]) O CONTROLLO DI PARITÀ SUI DATI, VELOCITÀ (→ AUMENTO DELLE PRESTAZIONI) DISTIBUENDO I DATI SU VARI DISCHI [STRIPPING] DEDICANDO ACCESSO IN PARALELLO AI DATI IN LETTURA)



3

LA SALA SERVER

SERVER

- I SERVER SONO RUROSI, INGOBLANTI, CALDI E AFFAHMATICI (DI ENERGIA)
- SONO UNA RISORSA PER L'AZIENDA: VANNO TUTELATI
- SPESO OSPITATI ALL'INTERNO DI ARMOADI RACK
 - ALCUNI PRE-DOTATI DI CONDIZIONAMENTO
- LA SALA SERVER È UNA STANZA DEMINUTATA DA GROSSE PARETI, POSSIBILMENTE INSONORIZZATA E ISOLATA TERMICAMENTE
- POGGIA SU 4 IMMANTI: CONDIZIONAMENTO, ELETTRICO, ANTI-INCENDIO, RETE
 - A SECCO (CON ARGON-130)
 - SE SERVONO 12.000 BTU, NEGLI ARMOADI ANDEBBERE 2 CONDIZIONATORI DA ALMENO 12.000 BTU (REDUNDANCY)
 - BEST-IN-TERMAL UNIT

↳ DÈNE AVERE UNA ADEGUATA CABLAGGIO DI RETE : DEVE ESSERE ANCHE POSSIBILE AI CAMBIAMENTI (BUONO BISOGNA TENERE I FILI IN ORDINE)

UPS : UNINTERRUPTIBLE POWER SUPPLY : (OPPURE GRUPPO DI CONTINUITÀ) → VA A BATTERE

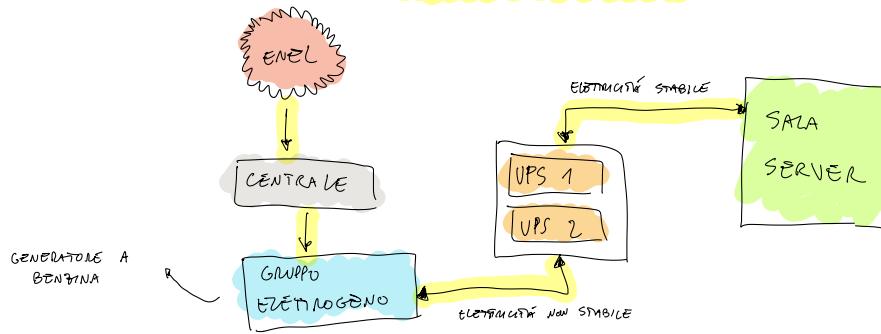
↳ DISPOSITIVO CHE PERMETTE DI MANTENERE LA TENSIONE STABILE E DI CONTINUARE A FORNIRE ENERGIA IN CASO DI BLACK-OUT (BRONI DI UN PAIO DI ORE , AVVIMENTI SERVONO IL GRUPPO ELETROGENO)

↳ SISTEMA DI MONITORING

GRUPPO ELETROGENO : DISPOSITIVO CHE CONverte L'ENERGIA CHIMICA DEL COMBUSTIBILE IN ENERGIA ELETTRICA UTILIZZABILE TRAMITE UN MOTORE INTERNO ED UN ALTERNATORE

↳ È SPESO UTILIZZATO COME FONTE DI ENERGIA DI EMERGENZA IN CASO DI INTERRUZIONI DELLA RETE ELETTRICA , AZIONANDO UNA TURBINA CHE GENERA ELETTRICITÀ

SCHEMA SALA SERVER ZETA



4

NETWORKING

• MEZZO O CANALE TRASMISSIVO : SUPPORTO FISICO TRAMITE IL QUALE UN SEGNALE SI PROPAGA DA UN PUNTO ALL'ALTRO IN RETE

• QUANDO SI METTONO IN COLLEGAMENTO DUE INTERLOCUTORI SI DICE CHE FRA QUESTI SI STABILISCE UN CANALE DI COMUNICAZIONE

• SU UN MEZZO TRASMISSIVO VI POSSONO ESSERE SIMULTANEAEMENTE PIÙ CANALI E, A SUA VOLTA , UN CANALE PUÒ USARE PIÙ MEZZI TRASMISSIVI

• CLASSIFICAZIONE :

↳ METTI DI TRASMISSIONE : (CANO DI DOTE) DOPPIO TELEFONICO , CAVO COASSIALE , FIBRA OTTICA , ETHERE (WIFI)

↳ CANALI : SIMPLEX (PUSSO DIRE UNIDIREZIONALE)

HALF-DUPLEX (PUSSO ADATTARSI IN 2 DIREZIONI)

FULL-DUPLEX (PUSSO SIMULTANEO IN 2 DIREZIONI)

PUNTO-PUNTO E MULTIPUNTO

TRANSPORTO SEGNALE
AUDIO - VISIONE
(PIÙ DI VETRO) → SPAZIO ELETROMAGNETICO ATTIVATO DAL QUALE VIAGGIANO LE Onde RADIO UTILIZZATE PER LE COMUNICAZIONI WIRELESS

• MEZZI TRASMISSIVI : PARAMETRI

bit x secondo

↳ VELOCITÀ DI TRASMISSIONE : DATI CODIFICATI IN BIT , SI MISURA IN BPS ED I SUOI MULTIPLI

↳ LARGHEZZA DI BANDA : MASSIMA CAPACITÀ TRASMESSIVA DI UN MEZZO , SI MISURA IN BPS

↳ CONFRONTO BANDA/VELOCITÀ : SU UN MEZZO TRASMISSIVO POSSONO ESSERE REALIZZATI PIÙ CANALI CHE SI "DIVIDONO" LA BANDA

→ DA DUE A OTTO F/LI INTRACCIAZI

SERVONO TANTI RIFRONTI

• CONNESSIONI FISICHE : DOPPIO TELEFONICO (RANO : RESISTENTE , BUONA VELOCITÀ MA RAPIDA ATTENUAZIONE DEL SEGNALE)

FIBRA OTTICA (PIÙ DI VETRO : MENO RESISTENTE , ALTISSIMA VELOCITÀ E LENTA ATTENUAZIONE DEL SEGNALE)

• CONNESSIONI WIRELESS : WI-FI (ETHERE COME MEZZO TRASMISSIVO , ECONOMICO E COMODO PER VIA DELL'ASSenza DI CABLAGGIO , MA MENO SICURO (PER ATTACCHI HACKER))

L1-F1 :

LIGHT FIDELITY ; LE FREQUENZE OCCUPATE PER LE COMUNICAZIONI APPARTENGONO ALLO STESSO SPECTRUM DELLA LUCE VISIBILE

↳ ASSSENZA DI INTERFERENZE RADIONEGLISTICHE

↳ IL SEGNALE È LIMITATO ALLA PORTATA OTICA , IL CHE RIDUCE I PROBLEMI DI SICUREZZA CAUSATI DALLE INTERCETTAZIONI

↳ SI BASA SULL'INVIO DI DATI ATTRAVERSO LA LUCE ALTERNANDONE LA FREQUENZA TRA 400 E 800 THz .

→ PIÙ LAN SONO DENTRO AD 1 MAN

→ ES. INTERNET , O CONNESSIONE GLOBALE

• "GRANDETTA" DUE CONNESSIONI : LAN (LOCAL AREA NETWORK) , MAN (METROLINK AREA NETWORK) , WAN (WIDE AREA NETWORK)

• TCP/IP (\Rightarrow TRANSMISSION CONTROL PROTOCOL / INTERNET PROTOCOL)

↳ TUTTE LE MODERNE RETI (LAN/MAN/WAN) SI BASANO SULL'INSIEME DI PROTOCOLLI O STACK TCP/IP

DEF. PROTOCOLLO: INSIEME DI REGOLE CHE DEFINISCONO IL FORMATO DEI MESSAGGI CHE DEVE ESSERE UTILIZZATO NELLA COMUNICAZIONE TRA DUE SISTEMI

• I DATI, PER TRANSITARE SU UNA RETE, DEVONO ESSERE DIVISI IN PICCOLI PEZZI, OGNUNO DEI QUALE VIENE INVIAZO SEPARATAMENTE.

↳ SU UNA RETE IP, QUESTI PEZZETTI SONO CHIAMATI "PACCHETTI"

• LO STACK TCP/IP È COSTRUITO SU UN MODELLO A STRATTI O UMBRE (SIMILE AL MODELLO OSI/ISO, NON RISPECCHIA LO STANDARD MA È SCRITO TACITAMENTE BENE ED È TALMENTE SEMPLICE DA ESSERE SEMPRE EFFICIENTE)

• IN UNA RETE TCP/IP OGNI HOST È IDENTIFICATO DA UN INDIRIZZO IP UNIVOCO

OSI/ISO

STACK TCP/IP

applicazione

applicazione

presentazione

con il suo
protocollo
(FTP, HTTP,
SMTP, ecc.)

sessione

protocolli TCP, UDP, ICMP, ...

trasporto

protocollo IP

rete

protocollo della
rete fisica
sottostante

collegamento dati

fisico

}

TCP/IP

→ MODELLO A STRATTI STACK TCP/IP

• DISPOSITIVI

↳ SCHEDA DI RETE: INTERFACCIA CHE PERMETTE DI COLEGARE UN HOST ALLA RETE

↳ DOTATA DI UN CODICE UNIVOCO (48 BIT) DETTO MAC ADDRESS

↳ MEDIA ACCESS CONTROL ADDRESS

↳ SWITCH: HA UNA TABEUA DI CORRISPONDENZA TRA PORTA E MAC ADDRESS DUE SCHEDE DI RETE AD ESSERE CONIFICATE

↳ EVOLUZIONE DEL KUB, CHE NON HA CORRISPONDENZE HA INVIATO LA MESSAGGIO TUTTO A TUTTI

↳ ROUTER: INSTRADARE PACCHETTI IN/OUT RISPETTO ALLA LAN

• SWITCH: INSTRADARE IL TRAFFICO ALL'INTERNO DI UNA LAN UTILIZZANDO MAC

ROUTER: INSTRADARE IL TRAFFICO TRA RETI DIVERSE (ES. LAN E INTERNET) UTILIZZANDO INDIRIZZO IP

• ACCESS POINT: OFFRE CONNETTIVITÀ WIRELESS ALL'INTERNO DI UNA RETE LOCALE,

↳ NON SI OCCUPA DI INSTRADAMENTO DEL TRAFFICO BENSÌ DI CONNETTIVITÀ.

• IPv4: IP VERSIONE 4

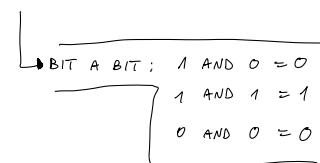
↳ 32 BIT (4 GRUPPI DA 8 BIT) RAPPRESENTAZIONE DECIMALE (ES. 157.138.22.21)

↳ DISTINZIONE DI 2 PARTI: INDIRIZZO DI RETE E INDIRIZZO DEL NODO

↳ LA MASCHERA DI RETE (USATA PER DISTINGUERE LA PARTE DELL'INDIRIZZO CHE IDENTIFICA LA RETE) È UN INDIRIZZO CHE VENECE ABBINATO ALL'INDIRIZZO IP (INDICA QUALI BIT DELL'INDIRIZZO IP INDICANO LA RETE DI APPARTENENZA)

↳ L'INDIRIZZO DI RETE SI OTTIENE COME RISULTATO DELL'OPERAZIONE AND TRA L'INDIRIZZO IP E LA SUA NETMASK

$$\text{ES. } 157.138.22.21 \text{ AND } 255.255.255.0 = 157.138.22.0$$



CLASSI DI IPv4 → DIVIDERE LO SPAZIO DI INDIRIZZAMENTO

• DAI PRIMI BIT DELL'IP POSSIAMO DETERMINARE LA SUA CLASSE E, DI CONSEGUENZA, LA SUA SUBNET MASK

	0	7	15	23	31
CLASSE A	0	NET-ID		HOST-ID	
CLASSE B	1 0		NET-ID		HOST-ID
CLASSE C	1 1 0		NET-ID		HOST-ID
CLASSE D	1 1 1 0		MULTICAST ADDRESSES		
CLASSE E	1 1 1 1 0			RESERVED	

- 1.0.0.0 – 126.255.255.255
(127.0.0.0 indirizzi di loopback)
 - 128.1.0.0 – 191.254.255.255
 - 192.0.1.0 – 223.255.254.255
 - 224.0.0.0 – 239.255.255.255
 - 240.0.0.0 – 254.255.255.255

→ MULTICAST E NON SINGOLI HOST → COME SE N HOST POSSANO SOLO 1

 NON HA LA SUBNET MASK PERCHÉ NON VENGONO UTILIZZATI PER IDENTIFICARE SINGOLI HOST ALL'INTERNO DI UNA RETE, MA PER IDENTIFICARE LA RETE STessa DOVE AL SUO INTERNO CI SONO VARI host (6410)

- LE DUE NOTAZIONI ATTRAVERSO LE QUALE È POSSIBILE INDICARE UN INDIVIDUO IP

l. INDICANDO ESPRESSAMENTE LA SUBNET MASK

- CLASSE A : 49.22.5.3 / 255.0.0.0
 - CLASSE B : 172.16.20.5 / 255.255.0.0
 - CLASSE C : 192.168.15.4 / 255.255.255.0

Indicando i bit attivi che compongono la subnet mask

- CLASSE A : 43.22.5.3 / 8
 - CLASSE B : 172.16.20.5 / 16
 - CLASSE C : 192.168.95.9 / 24

• ESEMPIO: classe C → SCOPO: TROVARE INDIRETTO DI RETE, CAPIRE QUANTI HOST SONO DISPONIBILI IN QUESTA RETE E QUALI (valori)

LA VENDO L'IP 192.168.1.50

- DETERMINARE LA CLASSE: PRIMO OTTOETO: 192 → APPARTIENE ALLA CLASSE C ($>= 192$ AND $<= 223$)

SUBNET MASK : DEFAULT : 255.255.255.0 → PRIMI 24 BIT ($8+8+8$) SONO RISERVATI ALLA RETE. GLI ULTIMI 8 ($x.x.x.x$) SONO PER GLI HOST.

- CONVERGENZA IN BINARIO DI ENTRABI (INDIRIZZO IP E MASK).

$$(1P) \text{ L } 192.168.1.50 \rightarrow 2^7 + 2^6 + 2^5 + 2^3 + 2^0 + 2^5 + 2^4 + 2^1 \rightarrow 11000000.10101000.00000001.00110010 = X$$

(NET) | 266 265 255 6

$$\text{WNA22-N}_{\text{NET}} = \sum_{i=1}^n A_i U_i V_i + \sum_{i=1}^m B_i U_i V_i$$

↳ 192-162-1-2 (PREEVUE 192-162-1-2/2)

INDICATED BROADCAST : ULTIMA UNANIMI DISSEMBUS VERA CORDIS - OPEN SOURCE OR CAN SUPPORT MASK UNKNOWN

2011-08-11

$$\begin{array}{r}
 \text{---} \\
 | \quad \text{OR} \quad 0 = 1 \\
 | \quad \text{OR} \quad 0 = 0 \\
 00000.0000000.11111111
 \end{array}$$

147000000 1010/000 00000001 111111

= 192 . 168 . 1 . 255 = INDIVIDUAL BROADCAST
(ULTIMATE INDIVIDUAL)

INDIA TO BURMA
(URMO INDIA)

Nº HOST ALIMENTO: Nº MARCA - UNID. A: 200 - UNID. A: 100

$$\frac{1}{= 2^8 - 1 - 1 = \textcircled{254}} \quad 192.168.1.255 \in \text{INDIANAPOLIS BROADCAST}$$

- HOST DISPONIBILITÀ / VADU = DA 192.168.1.1 A 192.168.1.254
 (192.168.1.0 È INDIVISIBILE IN RETE)

* QUINDI AD ESEMPIO L'HOST CON INDIRIZZO IP 192.168.1.50 È UN SERVER VAIANO ATTIVATO IN QUESTA DATA

$\int_0^{\infty} \left(\text{INDIRIZZO} \right) P \cdot D \pi^c(N) dN$

NAS E SAN

- **NAS : NETWORK ATTACHED STORAGE** (ARCHIVIAMENTO E CONDIVISIONE DEI DATI SU UNA RETE LOCALE)

La FLESSIBILITÀ: AUMENTARE LO STORAGE IN UN SISTEMA DI RETE ATTENDERE PUÒ NON ESSERE SEMPRE

- **DAS (DIRECT ATTACHED STORAGE)** E **SAS (SERVER ATTACHED STORAGE)** → PER AGGIUNGERE MEMORIA, BISOGNAVA ACQUISTARE NUOVI SERVER DA AGGIUNGERE O SOSTITUIRE, QUINDI BISOGNAVA SEMPRE Cambiare man mano L'INTERA STRUTTURA E SPesso INTERRUPPIRE IL SERVIZIO. ARCHIVIAMENTO NON COLLEGATO ALLA RETE, MA AD UN SINGOLO HOST/PC/SERVER E LA CONNESSIONE POTEVA AVVENIRE SOLO ATTIVAMENTO INTERFACCIA FISICO (USB, SATA...)

- **NAS**: ARCHIVIAMENTO È CONNESSO ALLA RETE → AUMENTO STORAGE SENZA TOCCHARE SERVER ORIGINARIO, MA COLLEGANDO ALTRA ARCHIVIAMENTO ALLA RETE

↳ COME FALE? AGGIUNGO UN NUOVO DISPOSITIVO NAS ALLA RETE SENZA TOCCHARE IL PRECEDENTE (POSSO ANCHE ESPANDERLO SENZA INTERRUZIONE IL SERVIZIO)

- COS'È FISICALMENTE? È UN COMPUTER (HOST) CON: X CPU(S), Y HARD DISK IN RAID, 1 SCHERA HDTV, Z SSD DA UTILIZZARE COME CACHE, W SCHEDA DI RETE VDSL (X, Y, Z, W NUMERI AUTOMATICI) ED 1 SISTEMA OPERATIVO (SINOGY)

- PER COSA SI USA? STORAGE PER QUALSIASI COSA (ANALOGO BACKUP)

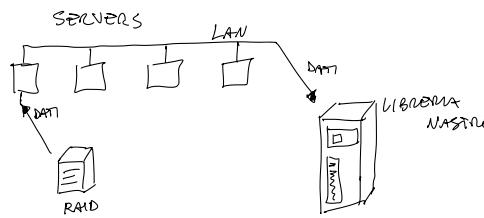
- NONOstante un NAS sia dotato di SISTEMA RAID E RIDONDANTO, È SEMPRE MEGLIO ESEGUIRE UN BACKUP DEI DATI (ANCHE DENTRO AD UN UNICO NAS)

- **SAN : STORAGE AREA NETWORK** (ENORME SAN ≈ CLOUD)

↳ DIVERSO DAL NAS, MA LI UTILIZZA COME COMPONENTI

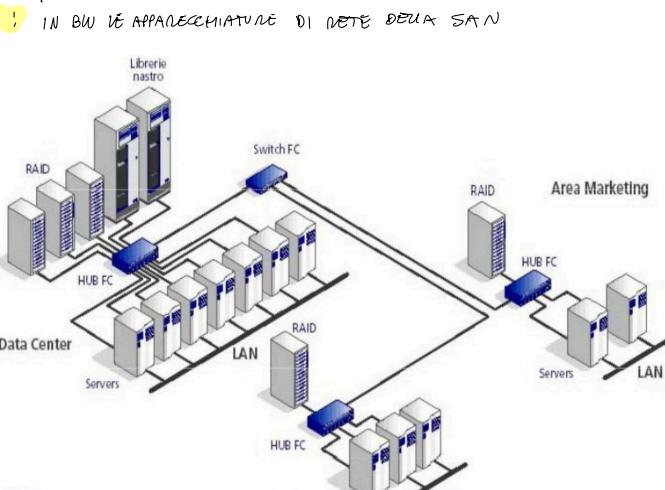
- DEF: PUÒ ESSERE PENSATA COME UNA LAN DEDICATA UNICAMENTE ALLO STORAGE, CHE È APPIANCATA ALLA LAN ATTENDENDO PER NON INFLUIRE SULLE LORO PRESTAZIONI

SISTEMA TRADIZIONALE:



SISTEMA SAN!

FC = FIBER CHANNEL



5b

VIRTUALIZZAZIONE

→ SIMULARE L'ESISTENZA DEL hardware (ESISTERE IN SISTEMI E PIÙ SO SU UN "UNICO" HOST)

- DEF: PROCESSO DI CREAZIONE DI UNA RAPPRESENTAZIONE "VIRTUALE" (BASATA SUL SOFTWARE) E' NON FISICA DI UN "OGGETTO"

- È POSSIBILE VIRTUALIZZARE: APPLICAZIONI, SERVER, DESKTOP, STORAGE, RETI

↳ È IL MODO PIÙ COMPLETO ED EFFICACE (ASSIMILE A DOCKER) PER RIDURRE LE SPESE IT AUMENTANDO AL TEMPO STESO L'EFFICIENZA E LA SCALABILITÀ

VIRTUALIZZAZIONE DEL SERVER

↳ COME ABBIANO VISTO LA CREAZIONE E MANTENIMENTO DI SERVERS SONO COSTOSI, COMPLICATI E NECESSITANO SFINTANTI AL 100%.

↳ LA VIRTUALIZZAZIONE RISOLVE QUESTO PROBLEMA: NON DOVER COMPRARE O MANTENERE COMPONENTI hardware MA UTILIZZARNE ONDULE SOFTWARE → CONSENTITO DI ESEGUIRE PIÙ SISTEMI OPERATIVI SU UN UNICO SERVER FISICO SOTTO FORMA DI MACHINES VIRTUALI

↳ È ALLA BASE DI QUALSIASI SERVIZIO DI CLOUD COMPUTING

VIRTUALIZZAZIONE DEL DESKTOP:

↳ COME SE SUDDIVIDESSE IL "GRANDE/CENTRALE" DESKTOP IN TANTI THIN CLIENT CHE AVVIANO UNA VERSIONE MINIMA DEL SISTEMA OPERATIVO

MACHINES LOCAL
HOSTS

↳ COLLEGAMENTO ATTRAVERSO DESKTOP REMOTO AD UNA MACCHINA VIRTUALE WINDOWS / LINUX

• 3 TIPI (PIÙ DIFFUSI) :

↳ 1) VDI : VIRTUAL DESKTOP INFRASTRUCTURE

↳ Gli utenti autorizzati possono accedere agli stessi server, file, app e servizi disponibili da qualsiasi dispositivo tramite un client desktop o un browser sicuro (quindi non tutti necessitano di un pc attivabile per accederlo → minor spreco di risorse)

↳ 2) RDS : REMOTE DESKTOP SERVICE

↳ USATO QUANDO BISOGNA VIRTUALIZZARE UN NUMERO LIMITATO DI APPLICAZIONI (NON L'INTERO AMBIENTE), QUESTO PERMETTE DI AVERE UN NUMERO DI UTENTI PIÙ ALTO

↳ 3) DaaS : DESKTOP - AS - A - SERVICE

↳ AFFIDA L'ATTIVITÀ DI VIRTUALIZZAZIONE DEL DESKTOP AI SERVICE PROVIDER, SIGNIFICA UTILIZZARE I SERVIZI DI UN PROVIDER ESTERNO PER OSPITARE E GESTIRE L'AMBIENTE DESKTOP (È IL NOSTRO) ANCHE FARLO INTERNALEMENTE

NOTA: PROVIDER: AZIENDA DI SERVIZI CHE DISPONE DI COMPUTER COSTANTEMENTE CONNESSI A INTERNET (TRAMITE LINEE SPECIALI)

• VIRTUALIZZAZIONE DELLA RETE

• CONSISTE NEGLA RIPRODUZIONE COMPLETA DI UNA RETE FISICA A NUOVO SOFTWARE

• OFFRANO LE STESSHE FUNZIONALITÀ DI QUELLA FISICA, MA ASSICURANO I VANTAGGI OPERATIVI E L'INDIPENDENZA DAL HARDWARE TIPICI DELLA VIRTUALIZZAZIONE

• SOFTWARE - DEFINED STORAGE : VIRTUALIZZAZIONE DELLO STORAGE

↳ APPROCCIO ALLO STORAGE CHE OFFRE UN MODELLO OPERATIVO PIÙ EFFICIENTE, ESEMPIO: GOOGLE DRIVE, ONE DRIVE, DROPBOX, ICLOUD... (SERVIZI CLOUD PaaS)

• MACCHINA VIRTUALE (VM)

↳ OGNI MACCHINA È COMPLETAMENTE INDEPENDENTE

• HYPERVISOR (STRATO SW): SEPARA LE VM DAL HOST ED ASSEGNA LE RISORSE DI ELABORAZIONE A CIASCUNA VM IN MODO DINAMICO

• PROPRIETÀ:

↳ PARTIZIONAMENTO: ESEGUIRE PIÙ SO SULLA STESSA MACCHINA FISICA SUDDIVIDENDO LE SUE RISORSE TRA LE VARIOLE VM

↳ ISOLAMENTO: OGNI VM GIRA IN UN AMBIENTE ISOLATO RISPETTO ALLA ALTRE VM E AL SERVER OSPITANTE

↳ PROTEZIONE DELLE PRIVACY ASSICURANDO UN FUNZIONAMENTO ADEGUATO AD OGNI VM

↳ INCAPSULAMENTO: POSSO INCAPSULARE TUTTO LO STATO DI UNA VM ED IL SUO STORAGE IN FILE.

↳ GRAZIE A LIVE MIGRATION POSSIAMO SPOSTARE UNA DA UN HOST A UN ALTRO SENZA DOWNTIME

INDIPENDENZA DAL HARDWARE: LE VM NON DIPENDONO SISTEMATICAMENTE DAL HW DEL SERVER FISICO OSPITANTE, MA DIPENDONO DAL HW ENVIATO DAL STRATO SW DEL HYPERVISOR

6

CLOUD → HERESSA "IN INTERNET" E NON LOCALE

• CLOUD COMPUTING (DEFINIZIONE):

↳ È LA DISTRIBUZIONE DI SERVIZI DI CALCOLO, COME SERVER, RISORSE DI ARCHIVIAZIONE, DATABASE, RETE, SW, ANALISI ECC.. TRAMITE INTERNET ("IL CLOUD")

↳ SOCIETÀ CHE OFFRONO IL SERVIZIO: PROVIDER DI SERVIZI CLOUD

• VANTAGGI:

↳ COSTO: IL COSTO DI HW E SW VIENE TUTTO RIVERSATO SUI FORNITORI DI SERVIZI CLOUD (CHE BISOGNA PAGARE). QUINDI SI OBTERRANNO LIBRI DI GESTIRE LO STORAGE/ARCHIVIAZIONE/ SW ECC... NOI DOBBIAMO SOLO PAGARE IL SERVIZIO FORNITO (CHE HA UN COSTO MINORE RISPETTO A MANTENERE SW E HW)

↳ VELOCITÀ: BASTA AGGIUNGERE/TOGLIERE FILE ANCHE IN GRANDI QUANTITÀ E DI GRANDI DIMENSIONI (IN BASE ALLO STATO VIRTUALIZZATO IL PREZZO CRESCE) CON POCHE CLICK

• 3 MODALITÀ DIVERSE DI DISTRIBUZIONE DEI SERVIZI CLOUD: PUBBLICO, PRIVATO O IBrido

↳ PUBBLICO: FORNISCE LE RISORSE DI CALCOLO (SERVER/ARCHIVIAZIONE) TRAMITE INTERNET

↳ ESEMPIO: MICROSOFT AZURE, AMAZON AWS, GOOGLE CLOUD...

↳ HW, SW E INFRASTRUTTURA DI SUPPORTO APPARTENGONO AL PROVIDER DEL SERVIZIO CLOUD, CHE LI GESTISCE

PRIVATO: PUÒ ESSERE DENTRO AL DATA CENTER O OSPITATO DA ALTRO (CLOUD PRIVATO GESTITO)

↳ È UN CLOUD IN CUI SERVIZI E INFRASTRUTTURA SONO GESTITI IN UNA RETE PRIVATA (ALL'INTERNO DELLA AZIENDA STESSA) O TRAMITE VPN

↳ CLOUD DEDICATO: CLOUD CONTENUTO IN UN ALTRIO CLOUD (ANCHE PUBBLICO) CHE SVOLGE SOTTO-CONTRACT

↳ MISTO: MISTO TRA PUBBLICO E PRIVATO

• **TIPI DI SERVIZI**

• **IaaS: INFRASTRUCTURE AS A SERVICE**:

↳ SI "AFFITTA" L'INFRASTRUTTURA IT, OVVERO SERVER E MACCHINE VIRTUALI, RISORSE DI ARCHIVIAZIONE, RETI E SISTEMI OPERATIVI

↳ IL PROVIDOR DI SERVIZI CLOUD GESTISCE L'INFRASTRUTTURA, NIENTE A NOI SPETTA L'INSTALLAZIONE E GESTIONE DELLE COSE "AFFITTATE"

• **PaaS: PLATFORM AS A SERVICE**:

SU RICHIESTA DI UTENTI / DEL MERCATO

↳ INSIEME DI SERVIZI DI CLOUD COMPUTING CHE FORNISCONO UN AMBIENTE ON DEMAND PER LO SVILUPPO, IL TEST, LA DISTRIBUZIONE E LA GESTIONE DI APPLICATION SOFTWARE

↳ AMBIENTE PIÙ FOCIAZATO SULLE APPLICATION (CREAZIONE E DEPLOYMENT DELLE APP)

↳ COME IaaS, PaaS INCLUSA L'INFRASTRUTTURA (NUOVI SERVER, ARCHIVIAZIONE E RETE) HA ANCHE MIDDLEWARE, STRUMENTI DI SVILUPPO, SERVIZI DI BUSINESS INTELLIGENCE (BI), SISTEMI DI GESTIONE DEI DATABASE È MOLTO ATTRAENTE

• **SaaS: SOFTWARE AS A SERVICE** (ESEMPIO: SERVIZIO DI POSTA ELETTRONICA BASATO SUL WEB (GMAIL, OUTLOOK..))

↳ È UN METODO PER LA DISTRIBUZIONE DI APPLICATION SOFTWARE TRAMITE INTERNET, ON DEMAND E IN GENERE IN BASE A UNA SOTTOSCRIZIONE

• **SCHEMA DEI TIPI DI SERVIZI**:

APPLICATION (SaaS)	→ FORNISCE APPLICATION PRONTE ALL'USO E TUTTO QUELLO CHE CI STA SOTTO (IaaS E PaaS)
PLATFORM (PaaS)	→ OFFRE UN HOSTING, FORNISCE AMBIENTE (PLATFORM) PER LO SVILUPPO/CREAZIONE DELLE APPLICATION (SOTTOALIMENTATO PER SVILUPPATORE)
INFRASTRUCTURE (IaaS)	→ HOSTING PER SITI WEB, APP ECC.. QUINDI SERVER, STORAGE E RETI VIRTUALIZZATI E OPERATI TRAMITE CLOUD

• **TERRAFORM**

↳ STRUMENTO CHE CONSENTE DI SPECIFICARE L'INFRASTRUTTURA DI NUBE ATTRAVESO UN LINGUAGGIO AD ALTO ULEVELLO (AUTOMAZIONE DELLA PARTE HARDWARE)

↳ PERMETTE DI INTERAGIRE CON TUTTI I SERVIZI CLOUD PIÙ RILEVANTI

↳ SISTEMA DI GESTIONE DELL'INFRASTRUTTURA → DEFINITA TRAMITE CODICE DI ALTO ULEVELLO (HCL = HASHICORP CONFIGURATION LANGUAGE)

↳ CREA UN GRAFO DELLE DIPENDENZE TRA I VARI ELEMENTI DELL'INFRASTRUTTURA

↳ ESEMPIO DI CODICE PER LA DICHARAZIONE DELL'INFRASTRUTTURA:

```
resource "aws_instance" "bastion" {
    ami           = "ami-007934628a8fcdec0"
    key_name      = "${aws_key_pair.bastion_key.key_name}"
    instance_type = "t2.micro"
    vpc_security_group_ids = ["${aws_security_group.bastion_sg.id}"]
    subnet_id     = "${module.vpc.public_subnets.0}"
    associate_public_ip_address = true
    user_data     = "${data.template_file.userdata_bastion.rendered}"
    tags = {
        Name = "${var.environment}-bastion"
        Environment = "${var.environment}"
    }
}
```

CODICE HCL

FILE SYSTEM

MEMORIA NON VOLATILE, PRESENTE ANCHE QUANDO IL
PC VIENE SPENTO

- DEF: LA PARTE DEL SISTEMA OPERATIVO CHE SI OCCUPA DELLA GESTIONE DEI FILE, FORMATTANDO OPPORTUNAMENTE LA MEMORIA DI MASSA, REGISTRANDO E LEGGENDO I FILE
 - 2 VISTE DIFFERENTI:
 - L 1) UTENTE: INSIEME DI FILE, DIRECTORY E SOTTO-DIRECTORY ED È USATO PER RICORDARNE E ORGANIZZARE I DATI IN MODO CHE SIANO FACILMENTE ACCESSIBILI AL SISTEMA ED AI SUOI UTENTI
 - L 2) OS: INSIEME DI STRUTTURE DI CONTROLLO E BLOCCHI DI DATI CHE COMPOSTO UNA PARTIZIONE (PORTELLA DEL DISCO).
 - Si CONCERNIA SULLA GESTIONE INTERNA E FISICA DEL FILE SYSTEM DA PARTE DEL SO
 - FILE
 - DEF: È UN'ABSTRAZIONE DEL SO CHE PERMETTE DI USARE IN MODO SENZIALE ED EFFICIENTE I DISPOSITIVI DI MEMORIA SECONDARIA
 - ATTRIBUTI: NOME, TIPO (ESTENSIONE), DIMENSIONE, PROTEZIONE (ACCESIBILITÀ (PERMESSI)), DATA, ora ...
 - IL NOME DEL FILE (COMPRESO IL PATH) DOVE ESSERE LUNGO 255/260 CARATTERI AL MASSIMO (LUNGHEZZA DIPENDE DA SO)
 - TIPI DI FILE:
 - FILE REGOLARI (es. .txt, .pdf...)
 - DIRECTORY → CONTENGONO FILE REGOLARI E NON
 - ORGANIZZATE IN UNA STRUTTURA GERARCHICA
 - FILE SPECIALI A BLOCCHI: (UNITÀ DI I/O A BLOCCHI)
 - RAPPRESENTANO DISPOSITIVI DI I/O CHE TRASFERISCONO DATI IN BLOCCHI
 - ES. DISCO RIGIDO: SO GESTISCE I DATI CHE LEGGONO/SCRIVONO IN BLOCCHI DI DIMENSIONI PESSE → RIDUCE OPERAZIONI DI I/O → ACCESSO EFFICIENTE
 - FILE SPECIALI A CARATTERI: (UNITÀ DI I/O A CARATTERI)
 - DISPOSITIVI DI I/O CHE TRASFERISCONO I DATI CARATTERE PER CARATTERE
 - DIRECTORY:
 - DEF: ENSEMBLE DI NOMI DI FILE (o/è ANCHE DIRECTORY) A CUI È ASSOCIAZIO UN NOME
 - FONDAMENTALE PER DARE UNA STRUTTURA GERARCHICA AL FILESYSTEM
 - OPERATORI: CREAZIONE, ELIMINAZIONE, RENOME, UNLINK, UNLINK...
 - STRUTTURA AD ALBERO: LA RADICE È DETTA RADICE DEL FILESYSTEM (/ o c:\)
- ### Esempi di strutturazione

The diagram shows three examples of directory structures:

 - a) livello unico:** A single level of files and directories.
 - b) due livelli:** Two levels of files and directories.
 - c) albero:** A tree structure with multiple levels of files and directories.

Legend:

 - file: represented by a green circle
 - directory: represented by a yellow square
- STRUTTURA DEL FILESYSTEM IN LINUX / UNIX
 - PROGRAMMA CHE VIENE ESEGUITO CON IL COMANDO DI STAMPALE NOMI DI FILE/DIRECTORY DEL CURRENT PATH
 - /etc CONTIENE FILE DI CONFIGURAZIONE DEL SISTEMA ES. DEKONI CHE OFFRONO SERVIZI
 - /bin CONTIENE ALCUNI PROGRAMMI ESSENZIALI (COME "ls")
 - ESSEMPIO DI PROGRAMMA ESEGUITIBILE "ls" (Scritto in BASH):
 - L ESEMPIO DI PROGRAMMA ESEGUITIBILE "ls" (Scritto in BASH):

```
#!/bin/bash
# Questo è uno script semplice che emula il comportamento di ls

# Ottieni la directory corrente
current_directory=$(pwd)

# Elencare i file e le cartelle nella directory corrente
for item in "$current_directory"/*; do
    # Controlla se l'elemento è una directory
    if [ -d "$item" ]; then
        echo "Directory: ${basename "$item"}"
    else
        echo "File: ${basename "$item"}"
    fi
done
```

Questa classe è chiaramente molto

più complesso

Possano però anche trovarsi
alcune di queste in qualsiasi
punto del sistema

- **/sbin** (SUPER USER BINARIES) : contiene programmi essenziali che l'utente comune non può e non deve lanciare automaticamente
- **/lib** : contiene le librerie dinamiche necessarie ai programmi delle directory precedenti (potrebbe anche contenere librerie per programmi specifici per CTF/DATA)
- **/usr** : contiene tutti i programmi che non sono necessari nelle prime fasi di avvio del sistema

• STRUTTURA (STORICA) DEL FILESYSTEM IN LINUX/UNIX

- **/home** (HOME DIRECTORY) : contiene gli spazi di memorizzazione personale degli utenti
- **/root** : è la root directory dell'utente root
- **/var** : dati di tutte le applicazioni di uso generale del sistema (non del singolo utente)
- **/dev** : file che rappresentano i dispositivi presenti nel sistema (es. dischi o terminali)
- **/tmp** o **/var/tmp** : file temporanei
- **/var/log**

- contiene i log (REGISTRI DEGLI EVENTI) del sistema, gestiti direttamente dalle relative applicazioni o dal servizio syslog
- es. un server di posta è un insieme di 1 o più servizi detti DAEMONS

• ogni demone ha la sua configurazione in /etc/c nome demone

• scrive tutto questo che gli succede nei file di log dentro /var/log

• analizzarli con "less" o "tail"

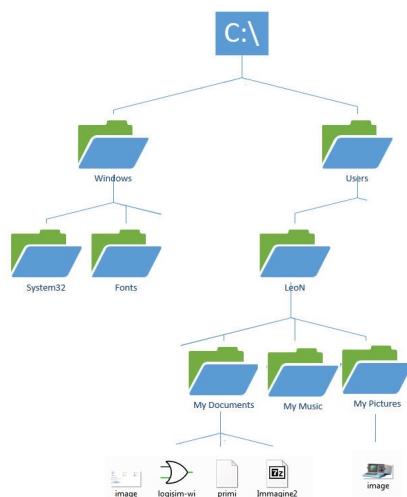
• con "tail -f" posso visualizzarne

anche durante la loro modifica

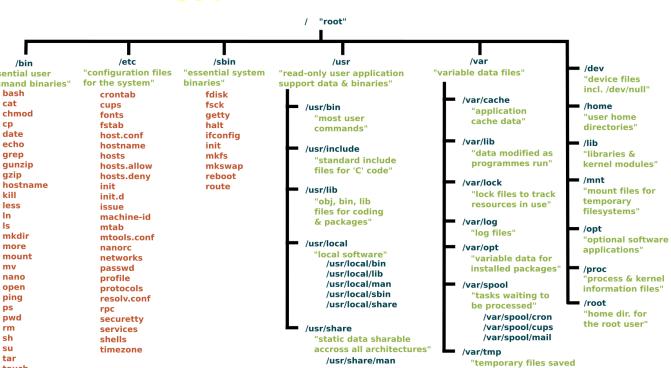
• STRUTTURA (ATTUALE) DEL FILESYSTEM LINUX

- **/boot** : file necessari al gestore d'avvio (boot loader → kernel e altri file di supporto)
- **/proc** : file o directory che rappresentano processi in esecuzione nel sistema (una directory per ogni processo)
- **/sys** : configurazioni a tempo d'esecuzione di alcuni aspetti del sistema collegati all'hardware
- **/run** (prima era /var/run) : piccoli file di stato delle applicazioni del sistema

FS WINDOWS



FS LINUX



- OGNI SO HA IL SUO FILESYSTEM PERSONALIZZATO:
 - **WINDOWS:**
 - **FAT32** → 1996
 - SISTEMA DI FILE A 32 BIT
 - L. WINDOWS 95 OSR2, WINDOWS 98
 - ***¹(FAT64)** → 2006
 - L. WINDOWS VISTA SERVICE PACK 1, WINDOWS SERVER 2008
 - **NTFS** → 1993
 - L. WINDOWS NT, WINDOWS 2000/XP/7/8/10
 - NEW TECHNOLOGY FILE SYSTEM
 - **MAC OS X:**
 - **HFS+³ (JOURNALED)** → 1998
 - L. MACOS SIERRA 10.13 E PRECEDENTI
 - **APFS⁴ (APPLE FILE SYSTEM)** → 2017
 - L. MACOS SIERRA 10.13 E SUCCESSIVE
 - **LINUX / UNIX - UKE**
 - **ext4** → 2006
 - L. KERNEL 2.6.19
 - DERIVA DA **EXT2** (SECOND EXTENDED FILESYSTEM) CHE A SUA VOLTA DERIVA DA **UFS** (BERKELEY FAST FILE SYSTEM)
 - FILESYSTEM DI DEFAULT LINUX
 - JOURNALED
 - FILE DI DIMENSIONI 16TB MAX E PARTIZIONI FINO A 16EB
 - **(WINDOWS)**
 - ***¹ exFAT: (EXTENDED FILE ALLOCATION TABLE)**
 - L. PENSATO APPositamente PER MEMORIE FLASH E PER DISPOSITIVI DI ARCHIVIAZIONE DA BASSA DIMENSIONE CONTENUTE
 - L. USATO QUANDO NTFS² RISULTA POCO FLESSIBILE
 - ***²NTFS:** PERMETTE DI GESTIRE 4 MILIARDI DI FILE E SI BASA SU UNA TABELLA **MFT** (MASTER FILE [ALLOCATION] TABLE)
 - L. SICURO, STABILE E FLESSIBILE
 - **(MACOS)**
 - ***³ HFS+ (come HFS MA JOURNALED)**
 - L. VECCHIO, OBSOLETO SPESO POCO PERFORMANTE HA PERPETTAMENTE INTEGRATO NEL SO CHE NE CORRE I DIFETTI
 - L. AFFIDABILE → AFFIDABILITÀ SI RIDUCE ALL'AVVENTURA DEL SO CHE SUPPORTA
 - ***⁴ APFS → APPLE FILE SYSTEM**
 - L. OTTIMIZZATO PER LE MEMORIE FLASH E A STATO SOLIDO, IMPLEMENTA UN MECCANISMO DI **COPY-ON-WRITE**
 - L. TECNICA DI OTTIMIZZAZIONE CHE MIRA ALLA RIDUZIONE DI OPERAZIONI DI DUPLICAZIONE DEI RISORSE DEL SISTEMA ATTIVANDO L'ELIMINAZIONE DI COPIE NON NECESSARIE

DEF: JOURNALED TO DO ! !

• • •