

1. Milyen jellemző paraméterei vannak egy mai asztali számítógépnek?

A mai számítógépek túlnyomó többsége a Neumann-elvek alapján működik. Számítógépek fő részei: központi egység, operatív memória, perifériák, háttértárolók és a sínrendszer. A számítógép fő hardveregységei: alaplap, processzor, RAM, winchester, videokártya, tápegység.

2. Mi a cache szerepe a mikroprocesszorban?

A cache (gyorsítótár), a processzorba vagy a processzor környezetébe integrált memória, ami a viszonylag lassú rendszermemória-elérést hivatott felgyorsítani úgy, hogy előzetesen beolvassa azokat az adatokat és programrészeket, amelyekre a végrehajtásnak feltételezhetően szüksége lehet.

3. Milyen a kettes komplementű számábrázolás?

1 biten 2 különböző érték (0,1). Egész számok ábrázolása - jellemzően 4 byte-on tároljuk. Egy komplementű ábrázolás esetén az első bit lesz az előjel. 1 bájtban így -127 +127 közötti számok ábrázolhatók. Kettes komplementű ábrázolás esetén egy nulla, -128 +127 közti számok ábrázolhatók egy bájtban.

4. Mit tud az UTF kódolásról?

Az UTF-8 kódolás lényege, hogy a 7 bites ASCII kódtábla karaktereit (angol kis- és nagybetűk, számok és gyakoribb írásjelek) az ASCII-kódjukkal jelöli, az egyéb karakterek kódját pedig "feldarabolja" és a darabokat egy vezérlőjelet követő több, egymás utáni bájtban helyezi el úgy, hogy a bájtok mindegyike 127 felett van (azaz így nem téveszthető össze a 7 bites ASCII-kódok egyikével sem).

5. Soroljon fel legalább 3 memóriatípust!

A számítógép gyors elérésű és dinamikusan változó adattároló egységét hívjuk a számítógép memóriájának. Fontosabb memóriatípusok: ROM (csak olvasható memória), PROM (programozható ROM), EEPROM (elektronikusan törölhető PROM).

6. Soroljon fel olyan hétköznapi eszközöket, amelyekben "számítógép" található!

Mobiltelefon, sütő, gáztűzhely, mikro, stb..

7. Mi a különbség egy szerver és egy kliens gép között?

A kliens olyan számítógép amely hozzáfér egy (távoli) szolgáltatáshoz, amelyet egy számítógép hálózathoz tartozó másik gép nyújt. Jellemzői: kéréseket, lekérdezéseket küld a szervernek, a választ a aszervertől fogadja, egyszerre általában csak kisszámú szerverhez kapcsolódik, közvetlenül kommunikál a felhasználóval.

A kiszolgáló vagy szerver olyan számítógépet, illetve szoftvert jelent, ami más gépek számára a rajta tárolt vagy előállított adatok felhasználását, a kiszolgáló hardver erőforrásainak (például nyomtató, processzor, háttértárak) kihasználását, illetve más szolgáltatások elérését teszi lehetővé.

8. Soroljon fel legalább 3 operációs rendszert!

Microsoft - Windows XP, Windows 7, Windows 8

Apple/Macintosh - Mac OS X

...

9. Mi a különbség az ssh és a telnet kapcsolat között?

Az SSH szolgáltatás és protokoll egyben. A Telnethez hasonlóan távoli számítógépek elérésére és operációs rendszerük vezérlésére fejlesztették ki 1995-ben. A Telnet és az SSH közötti alapvető különbség, hogy az SSH biztonságos kommunikációs csatornát épít ki a kliens és a szerver között, mégpedig úgy, hogy nyilvános kulcsú titkosítást használ a kommunikáló gépek hitelesítésére, és az átvitt adatok bizalmasságának biztosítására.

10. Milyen szolgáltatásai vannak a mai operációs rendszernek?

- kliens - szerver különbségek
- közös, osztott háttértár használata
- közös nyomtatási szolgáltatás használata
- szervizek kezelése
- levelezés, web, terminál elérés stb.
- hálózati szolgáltatások (DNS, DHCP, stb.)
- felhasználók kezelése
- információs adatbázis

11. Mit ért shell alatt? Nevezzen meg legalább kettőt!

Az operációs rendszernek kiadott parancsokat beolvasó és értelmező programokat az UNIX-terminológia parancsértelmezőknek, shelleknek (héjprogram, burok) nevezi. Ez a rendszerhéj az összekötő kapocs a rendszer magját képező kernel és a felhasználó között.

Unix alatt:

- Sh (Bourne shell)
- Ksh (Korn shell)
- Csh (C shell)
- Bash (Bourne again shell)

12. Mi az az alias? Hol találkozott vele?

Ha egy gyakran használt parancsot rövidebb formában akarjuk elérni:

alias elnevezés = "parancs"

Shellscriptnél és powershellnél találkozhatunk vele.

13. Milyen Unix fájlrendszer jellemzőket tud megemlíteni?

A Unix és a Unix-szerű operációs rendszerek minden eszközhöz egy eszköznevet rendelnek, de ennek nincsen köze ahhoz, hogy hogyan is érhetők el az eszközön lévő fájlok. Valójában a Unix létrehoz egy virtuális fájlrendszert, amelyben minden eszközön lévő minden fájl egy hierarchiába rendeződik. Ez azt jelenti, hogy a Unix-ban van egy gyökérkönyvtár, és minden létező fájl valahol ebben a gyökértől induló struktúrában helyezkedik el. Sőt, ennek a gyökérnek nem is kell tényleges fizikai hely. Nem kell az első merevlemezen lennie, még csak a gépben sem kell lennie, a Unix képes egy megosztott hálózati erőforrást gyökérkönyvtárként kezelni.

14. Milyen fájlrendszereket ismer?

Fájlrendszer típusok:

- lemezes fájlrendszerek
- adatbázis-fájlrendszerek
- tranzakcios-fájlrendszerek
- hálózati-fájlrendszerek
- speciális célú fájlrendszerek

15. Milyen fájlnev konvenciókat ismer Unix-ban?

- Név hossza nem korlátos
- Tetszőleges karakter használható
- Nincs kiterjesztés a Windows értelmezésében
- Ha a kezdőkarakter . (pont), akkor takart állományt hozunk létre

16. Milyen fájl jellemzőket ismer Unix-ban?

- Név
- Méret
- Létrehozás dátuma
- Tulajdonos

- Tulajdonos csoportja
- Hard link szám
- Jogosítványok

17. Magyarázza el a Unix-ban levő alapvető hozzáférési jogosultsági rendszert!

Alapvetően egy 3x3-as rendszer él (oktális rendszer)

Minden bejegyzésnek van

-Tulajdonos jogosultsága (u)

-Csoport jogosultság (g)

-Mindenki más jogosultsága (o)

Minden jogosultság három részből áll

-R - olvasási jog

-W - írási jog

-X - végrehajtási jog

20. Mit tud az idézőjelekről a Unix rendszerben?

“ ”: egyes karakterek vagy lefoglalt szavak speciális jellegét feloldják

‘ ’: megakadályozzák hogy a \$-t a változók jelzésének tekintse

` `: operátor, mely a két ` ` között levő jessort végrehajtja és a kimenetét adja vissza úgy, hogy az új sor - karakter helyére szóközt rak

21. Mit jelent az stdin, stdout?

Standard input - a billentyűzet, alapértelmezett bemenet

Standard output - monitor, alapértelmezett kiement

22. Hány szűrő kell egy csővezetékhez?

Kettő.

23. Mondjon példát arra, hol használhat reguláris kifejezéseket?

Szövegkereső, szöveg helyettesítő, szövegellenőrző feladatoknál alkalmazhatjuk őket.

24. Mi az ASCII kódtábla?

Szabványos amerikai kód. A kód jelkészlete az angol abc betűit, számokat, írásjeleket és vezérlő kódokat tartalmazza. Az ASCII jelkészlete 128 különböző szövegkaraktert a 0..127 előjel nélküli egész számokra képez le. ASCII-kódolást alkalmaznak a text-editorok (például a jegyzettömb) is.

25. Mik azok a környezeti változók?

A környezeti változók (angolul environment variables) a felhasználó bejelentkezésekor értéket kapnak, és a shellben bárholnan elérhetőek.

26. Adja meg, hogy a Unix-ban milyen típusú(ak) lehet(nek) a változó tartalma(k)!

Lehetnek speciális változók, környezeti változók.

27. Mit jelent a parancsbehelyettesítés?

` `: operátor, mely a két ` ` között levő jessort végrehajtja és a kimenetét adja vissza úgy, hogy az új sor - karakter helyére szóközt rak.

28. Sorolja fel, hogy milyen műveletek léteznek a Unix shellben?

-eq egyenlő

-ne nem egyenlő

-lt kevesebb mint

-le kevesebb mint vagy egyenlő

-gt nagyobb mint
-ge nagyobb mint vagy egyenlő
= két szöveg egyenlő
!= két szöveg nem egyezik
! a kijelentés eredményének negáltja

29. Melyik shell utasításnak van befejezési eredménye?

A sed parancs egy adatfolyamszerkesztő, aminek a segítségével egyszerű módosításokat végezhetünk a bemeneten érkező adatsorokon. Ha a sed szkriptben szereplő összes parancs (egy sorra több parancs is vonatkozhat) feldolgozta az aktuális feldolgozandó sort, akkor a soron végzett utasítások eredménye, az új sor tartalma kikerül a standard kimenetre.

30. Hogyan implementálják a logikai értékeket a Unix shellben?

Test paranccsal a szkriptben logikai ellenőrzést hajtunk végre. Ha lefut a parancs, rendben elvégzi a feladatát, akkor logikai igaz értéket ad vissza. (0) Ha nem tudja rendesen elvégezni a feladatát, üres sort olvas be, hamis értéket ad vissza. (1)

31. Lehet-e paramétereket kezelő függvényeket definiálni a Unix alatt?

Igen.

32. Tudja-e futtatni a végrehajtási jogosultság nélküli shell szkriptet?

Nem.

33. Mi az az IFS?

Internal Field Separator, az alapértelmezett elválasztó helyett új elválasztó karakter definiálása

34. Milyen feladatokat tud elvégezni a sed-del?

A sed parancs egy adatfolyamszerkesztő, aminek a segítségével egyszerű módosításokat végezhetünk a bemeneten érkező adatsorokon.

35. Írja le általánosan egy sed parancs szintaxisát!

sed 'szkript' fájlnev1 fájlnev2 ...

36. Mi a különbség a sed használatában a " és a ' idézőjel használata között?

" használatakor a változó értéke helyettesítődik be, ' használatakor nincs érték behelyettesítés

37. Jellemezze az awk lehetőségeit!

alkalmazhatjuk szöveges fájlkon való műveletek végzésére. Míg a sed sorokon belüli tevékenységei korlátozottak, nincsen sok aritmetikai lehetőség valamint vezérlési szerkezetek is hiányoznak, addig az awk ezekre megoldást kínál.

éigolvassa a bemenetet, azt feldolgozza, majd a kimenetre ír.

az awk is soronként veszi a bemenetet (ezt a bemeneti sort gyakran aktuális rekordnak is nevezi), azt a kívánalmaknak megfelelően átalakítja, és a kimenetre írja ezt az eredményt.

Azt az utasítást, ami alapján a soronkénti átalakítást végezzük, az awk parancs programjának nevezzük. Ezt a programot, ahogy a sed esetében is láttuk, fájlba is írhatjuk, és ezt afájlt futtathatjuk.

38. Adja meg hogy milyen parancsblokkok találhatók az awk-ban!

BEGIN, END

39. Használható-e az awk aritmetikai feladatok megoldására?

Míg a sed sorokon belüli tevékenységei korlátozottak, nincsen sok aritmetikai lehetőség valamint vezérlési szerkezetek is hiányoznak, addig az awk ezekre megoldást kínál.

40. Mi az MBR és mi a feladata?

Személyi számítógépeken (PC) a rendszertöltés (boot) folyamatának részeként a BIOS betölti a merevlemez fő rendszertöltő rekordját (Master Boot Record, MBR), innen folytatódik az operációs rendszer betöltése. A fő rendszertöltő rekordban legfeljebb négy partíció adatainak tárolására van hely, ezért a merevlemez legfeljebb négy valódi partíciót tartalmazhat. Particionáláskor meg kell adni az aktív (boot) partíciót, ami MS-DOS és Windows rendszereken rendszerint az első elsődleges partíció, hogy a rendszer bootolásra képes legyen.

42. Írjon le legalább egy UNIX-LINUX management lehetőséget!

General purpose Linux computing

Hardware Support

Older Platforms

Backups & Restores

43. Milyen hálózati kapcsolódási lehetőségeket ismer?

vezetékes (LAN), vezeték nélküli, mobil szélessávú, VPN, DSL

44. Mit ért csomagkapcsolt hálózat alatt?

Két alapvető összetevőből állnak: a kapcsolóelemekből és az átviteli vonalakból. Ebben az esetben a tetszőleges hosszúságú üzenetek meghatározott terjedelmu csomagokban érkeznek meg, mivel a csomag hossza maximálva van. Amennyiben a csomagkapcsoló hálózatban a csomagméretet meghaladó üzenetet kell átvinni, akkor a forrásállomás az üzenetet részekre tördeli, és az egyes részeket egy-egy csomag alakjában továbbítja. Az egyes üzenetdarabok így elszakadhatnak egymástól, és csak a célállomásnál áll össze belőlük a teljes egész eredeti üzenet. Egy csomagkapcsolásos hálózat egyidejűleg több üzenetet továbbít az átviteli vonalakon. Ezt az átviteli eljárást multiplexelés-nek nevezik. A csomagkapcsolás nagyon hatékonyan képes a vonalak kihasználására, mivel az adott két pont közötti összeköttetést több irányból érkező és továbbhaladó csomag is használja.

45. Mit ír le az OSI modell?

Az Open Systems Interconnection Reference Model, magyarul a Nyílt rendszerek összekapcsolásreferenciamodellje (OSI-modell vagy OSI-referenciamodell) egy rétegekbe szervezett rendszer absztrakt leírása, amely a számítógépek kommunikációjához szükséges hálózati protokollt határozza meg, s amelyet az Open Systems Interconnection javaslatban foglalt össze.

46. Nevezzen meg hálózati topológiákat!

busz, fa, lánc, gyűrű, csillag, horgolt

47. Mi a feladata egy switch-nek?

Csillag topológiájú hálózatban manapság használt kommunikációs útvonalakat kiépítő csomópont. A hálózatba való első bekötésekor ugyanúgy viselkedik, mint a HUB, de „figyeli” a kiküldött csomagok „sorsát”. Az így begyűjtött információk alapján a 2 későbbiekben már célzottan, mindig a megfelelő című géphez küldi a csomagot, így nem generál felesleges adatforgalmat.

48. Mi a feladata a routernek?

Útvonalválasztó (külső/belső hálózat). Leggyakrabban a helyi hálózat internethez való illesztésére használatos. Két címmel rendelkezik: a helyi hálózat felé látható belső címmel, melyen keresztül a LAN gépeivel kommunikál; illetve egy WAN-címmel, amelyen keresztül az internet többi szereplőjével kommunikál.

49. Hogyan jellemezné az IPv4 címeket?

IPv4 szerinti IP-címek 32 bites egész számok, amelyeket hagyományosan négy darab egy bájtos, azaz 0 és 255 közé eső, ponttal elválasztott decimális számmal írunk le a könnyebb olvashatóság kedvéért.

50. Hol találkozunk a DNS-sel az informatikában??

A Domain Name System (DNS), azaz a tartománynévrendszer egy hierarchikus, nagymértékben elosztott elnevezési rendszer számítógépek, szolgáltatások, illetve az internetre vagy egy magánhálózatra kötött bármilyen erőforrás számára.

51. Mi az a DHCP?

A dinamikus állomáskonfiguráló protokoll (angolul Dynamic Host Configuration Protocol, rövidítve DHCP) egy számítógépes hálózati kommunikációs protokoll.

Ez a protokoll azt oldja meg, hogy a TCP/IP hálózatra csatlakozó hálózati végpontok (például számítógépek) automatikusan megkapják a hálózat használatához szükséges beállításokat. Ilyen szokott lenni például az IP-cím, hálózati maszk, alapértelmezett átjáró stb.

A DHCP szerver-kliens alapú protokoll, nagy vonalakban a kliensek által küldött DHCP kérésekből, és a szerver által adott DHCP válaszokból áll.

52. Milyen szerver elérési módokat ismer?

TCP, FTP

53. Mire szolgál a HTTP protokoll?

A HTTP (HyperText Transfer Protocol) egy információátviteli protokoll elosztott, kollaboratív, hipermédiás, információs rendszerekhez.

A HTTP egy kérés-válasz alapú protokoll kliens és szerver között. A HTTP-klienseket a „user agent” gyűjtőnévvel is szokták illetni. A user agent jellemzően, de nem feltétlenül webböngésző.

A HTTP általában a TCP/IP réteg felett helyezkedik el, de nem függ tőle. A HTTP implementálható más megbízható szállítási réteg felett is, akár az interneten, akár más hálózaton.

54. Mi történik, ha a public_html könyvtárban nincs index.html fájl?

Minden webkönyvtár alapértelmezett fájlja az index.html, index.htm vagy index.php. Ez a gyakorlatban azt jelenti, hogy az oldalad látogatóit (<http://tedomainneved.hu>) a szerver valójában ide vezeti: <http://tedomainneved.hu/index.html>. Ez így működik az összes nyilvánosan elérhető könyvtárban, beleértve az aldomaineket is.

ha nincs index.html fájl ez nem történik meg.

55. Hogyan lehet jelszóval védeni egy weben lévő könyvtárat?

Néha szükségünk lehet arra, hogy a weboldalunkra feltöltött bizonyos mappát/mappákat jelszóval levédjünk. Ennek több megoldási lehetősége van, de a htaccess a legegyszerűbb. Ezt Apache szerver esetén bárki megteheti (legtöbb tárhely szolgáltatónál ez van).

56. Mit ért virtuális host alatt?

Képesse tesz egy egyedülálló gépet arra, hogy többszörös domainnel rendelkező web server legyen.

57. Mit jelent az SSI vagy CGI jog a webszervereken?

SSI: Szerver oldali beágyazás- egy olyan programra való hivatkozás, melyet a szerver a kért weboldal elküldése előtt lefuttat és eredményét a html dokumentumba küldés előtt beépíti.

58. Milyen Windows szkript lehetőségeket ismer? Van egyáltalán?

Python, c#, php,xml, nullscript, pascal, java

59. Mi biztosítja PowerShell-ben az "autoexec.bat" szerepét?

Az AUTOEXEC.BAT egy olyan fájl, ami a gép indulásakor végrehajtandó parancsokat tartalmazza. Ennek szerepét a PowerShellben az ún. profilok veszik át. A profilok egyszerű fájlban tárolt szkriptek, amelyek a PowerShell indításakor automatikusan lefutnak.

60. Hogyan biztosítják PowerShell alatt a biztonságos szkript futtatást?

A providereket úgy kell elképzelni, mint „szolgáltatókat”, amelyek egyfajta „távvezérlést” tesznek lehetővé bizonyos eszközökhöz. Ezek az eszközök lehetnek lokálisak, vagy távoli gépen futtatott szolgáltatások. Tehát nem közvetlenül az adott eszközöket vesszük igénybe, hanem a providert utasítjuk egy-egy utasítás végrehajtására. Azt, hogy ezt miképp hajtja végre, már nem a mi problémánk.

61. Milyen a PowerShell parancsok felépítése?

Cmdlet __ A paraméter neve __ A paraméter értéke

62. Soroljon fel PowerShellben legalább két változó láthatósági formát?

A PowerShell-ben 4 láthatósági szintet különböztetünk meg:

Global: a globális hatókör az egész PowerShell-re érvényes. Bármelyik munkamenetből elérhetőek.

Local: Egy adott függvény vagy szűrő hatáskörébe eső változók láthatóságát nevezzük lokálisnak.

Miután a helyi hatókör befejezte futását, a benne lévő változók elvesznek.

Script: Egy adott munkamenetben érvényes változók. A munkamenet befejeződésével a változók megsemmisülnek.

Private: A zárt hatókör megegyezik a lokális láthatósággal, azzal a különbséggel, hogy a gyermek hatókörökbe nem öröklődnek a belső hatókör tartalmába eső privát változók.

63. Hogyan irányítjuk át PowerShellben az outputot? Lehet?

Igen lehet pl.:

I) "Hajrá Fradi!" > fardi.txt # felülírás, új fájl

II) Get-Content fradi.txt # PS típus

Cat fradi.txt # unix típus

Type fradi.txt # dos típus

III) "Hajrá UTE!" >> fradi.txt # hozzáfűzés, ha nincs fájl, létrehozza

Itt nincs < vagy << átirányítás

64. Hol és mire használható a dot sourcing?

Amikor dot source-ingolsz egy szkriptet, az összes változó és funkció meg lesz határozva a szkriptben, és addig fennál a shellben, ameddig a szkript véget nem ér.

65. Mit jelent a PowerShell függvények nevesített paraméterezési lehetősége?

(Kiindulva abból, hogy a PS valójában C#Szkript) Nem szükséges betartanod azt, hogy egy függvénynek milyen paraméterei és milyen sorrendben következnek. Tehát ha például ha van egy a(b, c) függvényed, akkor te meghívhatod fordított sorrendben is, előbb a c, majd a b paraméterét megadva, persze ki kell mondanod, hogy mi mit jelent: a(c: 5, b: 9). Ekkor a megfelelő változónak a megfelelő érték lesz adva.

66. Mi a különbség a mikroprocesszor és mikrokontroller között?

A mikrokontroller egy mikroprocesszor kiegészítve az áramköri lapkájára integrált perifériákkal.

Régebben mikroprocesszor-típusokat használtak a vezérlési feladatok elvégzésére. A mikroprocesszor használatakor a szükséges perifériák miatt további IC-ket kellett beépíteni. Az áramköri technológia fejlődésével egyre több perifériát az IC tokba lehetett integrálni, így alakult ki a mikrokontroller, nagyon tömör áramkört eredményezve.

67. Mi a "Harvard architektúra" legfontosabb jellemzője?

A Harvard-architektúra egy számítógép-felépítési elv, amelyben a programkód és az adatok külön, fizikailag elkülönített útvonalakon közlekednek a processzor felé.

68. Mire használható a lebegőpontos számábrázolás?

A számítástechnikában a lebegőpontos számábrázolás lehetővé teszi a valós számok kezelését véges tárhely esetében, széles skálát fedvén le a számhalmazon belül. A számok rögzített számú számjegyekkel ábrázolhatóak, és egy kitevő (exponens) segítségével vannak skálázva. A skálázás alapja leggyakrabban 2, 10 vagy 16.

69. Mit jelent az asszimetrikus kódolás?

Ennél a kódolási eljárásnál két, egymást mindig kiegészítő kulcs létezik. Az egyik kulcs - a nyilvános kulcs (Public Key) - az üzenet kódolására, a másik - a titkos kulcs (Private Key) - a dekódolásra. A két kulcs együtt egy kulcspárt alkot.

A dologban az a különös, hogy az egyik kulcsból a hozzá tartozó másikat egyáltalán nem könnyű kitalálni vagy kiszámítani. Ezért a kulcspár egyik tagját nyilvánosságra hozhatjuk. (PL:két szám összeszorozás)

70. Mit jelent a bináris ftp lehetősége?

A fájlátviteli protokoll (FTP: file transfer protocol) a gépek közötti hálózati fájlátvitel segédeszköze. Lehetővé teszi a fájlok mozgatását a kliens és szerver között mindkét irányban, könyvtárak létrehozását, átnevezését, illetve törlését.

Bináris FTP-kapcsolat esetén az átküldött fájlok bájt helyesen, változtatás nélkül érkeznek meg.

71. Mit jelent a szöveges ftp? Létezik egyáltalán?

Szöveges kapcsolat esetén az FTP gondoskodik az eltérő szöveges fájlformátumok átalakításáról (például a DOS–UNIX szövegfájlok konverziójáról). A helyes átviteli mód kiválasztása az átvitel előtt a felhasználó feladata. Alapértelmezésben az FTP a szöveges módot használja, ami sok hibára ad lehetőséget.

72. Hogyan irányíthatja át a szabványos bemenetet PowerShellben?

pipeline annyit jelent, hogy egy adott commandlet kimenete (tehát a parancs után tesszük a | vonalat, ami magyar kiosztású billentyűzeten az AltGr-W kombinációra hozható elő) lesz a következő parancs bemenete (a következő parancs az előző parancs által kiadott adatokkal fog dolgozni).
Get-ChildItem | Out-File out.txt

74. Mire használható a profile.ps1 állomány? Van a UNIX-ban megfelelője?

Minden indításkor automatikusan lefuttathatóak parancsok, scriptek stb. Az UNIX megfelelője a .profile állomány

75. Mit értünk PowerShell modulon?

A PowerShell a moduláris felépítésének köszönhetően nagyon sok funkcióval bővíthető. Ezeket a funkciókat modulok vagy providerek formájában tudjuk használni a legegyszerűbben. Úgy kell ezt elképzelni, hogy például van egy új programunk, ami valamilyen szolgáltatást valósít meg. Ekkor lehetőségünk van egy olyan modult írni hozzá, amelyen keresztül képesek vagyunk kezelni (mi vagy épp mások) a PowerShell alapvető parancsainak a segítségével. Például új felhasználókat tudunk felvenni, adatbázisokat manipulálni, vagy csak egyszerűen egy távoli gép beállításait kezelni, frissíteni.

76. Elég-e a core PowerShell modul a registry módosításához? Miért?

Nem, mert core feladata: Szokásos grafikus lehetőségek (File, Edit, View) • Tools – munka ablak font, szín beállítások. • Debug – Szokásos nyomkövetési lehetőség! Kiegészítés szükséges.

77. Hogyan használhatja PowerShellben a parancsbehelyettesítést?

`parancs` parancsot végrehajtja, parancs kimenete kerül a helyére

- Bash shellben: \$(parancs) a=`date` ; b=`date` # • echo \$a # ??? A date szó lesz az eredmény.

78. Hogyan készíthet ciklust sed scriptben?

cat osztaly | sed 's/3/9/g ; s/9/21/' # Cseréljük az osztály soraiban az összes 3-at 9-re, majd az első kilencest 21-re minden sorban!

79. Jellemzően milyen állományokat talál az /etc könyvtárban?

A FHS alapján a /etc mappába az aktuális gép szintjén (rendszer szinten) érvényes, statikus konfigurációs állományok (hosztnev, ifconfig, dns resolution, webszerver hosztok, ssh security) kerülnek.

80. Mire szolgál a hálózati csomagok TTL adata?

A Time To Live (TTL) számítógép-hálózatok működésével kapcsolatos fogalom. Szó szerinti jelentése: élettartam. A TTL mezőre azért van szükség, mert nélküle redundáns hálózatoknál hibás beállítás esetén előfordulhatna, hogy az IP csomagok körbe-körbe keringenek. A TTL mező nélkül a hálózatban a végtelenségig keringő csomagok jöhetnének létre és használhatatlanná válna mivel folyamatosan foglalnák a hálózat sávszélességét.

80. Mire szolgál a hálózati csomagok TTL adata?

A TTL mezőre azért van szükség, mert nélküle redundáns hálózatoknál hibás beállítás esetén előfordulhatna, hogy az IP csomagok körbe-körbe keringenek. A TTL mező nélkül a hálózatban a végtelenségig keringő csomagok jöhetnének létre és használhatatlanná válna mivel folyamatosan foglalnák a hálózat sávszélességét.

81. Mit mutat meg a "Netmask"?

"hálózati maszk"-ra. (netmask) A netmask ugyanúgy néz ki mint egy IP cím, tehát 4 pontokkal elválasztott decimális szám. A hálózati maszk azonban azt mutatja meg, hogy egy adott alhálózaton, az elejétől fogva, hány bitnek kell megegyeznie az IP címben. Ha például az első 24 bitnek kell megegyeznie, akkor az alhálózati maszk a következő lesz:

Mask: 255.255.255.0 – megadja a LAN méretét (256)

Netmaszk

82. Mit értünk "nem routolható" IP címen?

az előtag mező értéke a bináris 1111111010. Ezt 54 nulla követi, aminek következtében a kapcsolati szintű címeknél az összes hálózati előtag ugyanaz, ezért nem routolhatóak.

83. Mire szolgál a "gateway"?

- Gateway IP: a kivezető út IP címe (router)

84. Mi az "ARPANET" és milyen lehetőségeket teremtett?

A hálózat ősprjektje: ARPANET. NCP a kezdeti ARPANET kommunikációs szabvány.

ARPANET szolgáltatások • File transfer (FTP – RFC354, 1971,73) • Terminál szolgáltatás (telnet – RFC 137, 1971, RFC854) • Erőforrások megosztása (NFS) • Üzenetek továbbítása (Mail- RFC524,561 1971,73) • Hang továbbítás (NVP) – nem sikeres, ma helyette: VOIP!

85. Mi az IPv6? Miért van rá szükségünk?

- Születésének fő oka: IPv4 előrelátható szűkössége!
- IPv6 128 bites címeket használ!
- 8 darab 16 bites szám hexa alakja:
- 2015:0a0d:0102:1961:0324:fe01:03ab:0405
- Első 64 bit: subnet prefix

Második 64 bit: interfész azonosító

- Első standard: RFC2640 (1998 dec.)

86. Mondjon példát a "setuid" bit hasznosságára!

Futtatható fájl mindig a tulajdonosa, illetve csoportja jogosultságaival fusson. Ez a gyakorlatban az effektív uid/gid értékeket állítja, a program a setuid/setgid rendszerhívásokkal visszatérhet a való uid/gid használatára.

87. Milyen célt szolgál az ACL használata Unix-Linux rendszerben?

- Hozzáférés-szabályozási listák (Access Control List, ACL)
- (minden állományhoz egyenként adhatunk felhasználókat különböző jogokkal!)

88. Létezik Windows rendszerben az ACL lehetősége?

Nincs.

89. Mi az analóg- digitális jelek közti alapvető különbség?

Analóg jel, információ, folytonos jelértékek! A környezeti paraméterek, távolság, hőmérséklet, zene, zaj, áramerősség stb. természetes értékei! • Digitális jel, információ, diszkrét, nem folytonos értékek tárolása!

90. Mi az adat, cím, vezérlő sín feladata?

vezérlő sín: lehetővé teszi adatok vagy tápfeszültségek továbbítását a számítógépen belül vagy számítógépek, illetve a számítógép és a perifériák között.

adat: Az adat elemi ismeret. Az adatokból gondolkodás vagy gépi feldolgozás útján információkat, azaz új ismereteket nyerünk.

cím: egyértelműen azonosítja

91. Hogyan készíti szűrőt UNIX illetve Powershell alatt? Lehet?

Fontosabb kész szűrők: • cut, tee, sort, uniq, wc, grep

Szűrő formában |

Szűrő példa: • cat nevsor | grep Pista # Eredményül kapjuk a Pista-t tartalmazó sorokat.

92. Mi a lényegi különbség a UNIX ls és a PowerShell Get-ChildItem parancsának eredménye között?

Mind az aktuális könyvtár tartalmát listázza ki, csak az ls nem írja ki a jogosultságokat és a utolsó módosítás dátumát.

93. Melyik környezetben tud használni reguláris kifejezést a UNIX, a Windows PowerShell világában vagy mindkettőben:

Mindkettőben.

94. Milyen speciális jelentése van annak, ha egy fájlnev .-tal kezdődik?

Ha a kezdőkarakter .(pont), akkor takart állományt hozunk létre.

95. Mikor használhatóak jól a reguláris kifejezésekben létrehozható csoportok és miért?

? Egy zárójelbe () tett reguláris kifejezés. Az a szöveg felel meg neki, ami a zárójelben szerepel. Egyfajta csoportosító szerepe van. Például ezekre a csoportokra a sed programban lehet a sorszámmal hivatkozni. Az ?(alma|dió)fa? reguláris kifejezésnek az ?almafa? és a ?diófa? szöveg felel meg.

96. Mi a lényegi különbség a UNIX shell és a PowerShell csővezetéken áthaladó adatok között?

Powershell : több parancsot kell egymás után kiadnunk, és ezeknek a parancsoknak az eredményével szeretnénk tovább dolgozni.

Unix: egyik program kimenetét a másik program bemenetével köti össze.

. Unix shell text alapu, tehat ha pipeolsz akkor csak a szoveg megy at. Powershell objektum alapu, pipenal mindenfele tulajdonsagot kulon tudsz lekerdezni az objektumbol, ergo tudsz groupolni is meg barmit smile hangulatjel

98. Milyen kiterjesztésűnek kell lennie egy PowerShell és egy shell szkriptnek? Van egyáltalán előírás vagy szabadon megválasztható?

PowerShell szkriptek .ps1 (bár nem vagyok benne biztos, hogy ki van kényszerítve... megfelelő társítás esetén lehet, hogy a .fradi is fut!), Linux esetén -úgy mint semmi más fájl esetén sem Linux alatt- semmi nincs kikényszerítve... általában nem is használják, de ha nagyon muszáj, .sh-nak szokás elnevezni. Linuxnak a shebang line kell a fájl elejére, hogy a shell tudja, hogy neki most shell (és hogy milyen shell) szkriptet kéne futtatnia.

99. Lehet paramétere egy szűrőnek? Ha igen, adjon meg egy tetszőleges példát, ha nem, magyarázza meg, hogy miért nem lehetséges!

Igen, cat nevsor | grep Pista # Eredményül kapjuk a Pista-t tartalmazó sorokat

100. Mi a különbség az stdout, stderr csatorna között? Léteznek PowerShellben?

stdout (1) - monitor, alapértelmezett kimenet

- stderr (2) – monitor, alapértelmezett hibakimenet

Nemes egyszerűséggel annyi, hogy két külön fájlcsatorna. Az stdout-ra a szabványos kimenetek (tehát minden olyan érdemi kimenet és felhasználói interakció) mennek, az stderr-re pedig a hibák. Ez két külön stream, külön módon kaphatóak el és értékelhetők ki. (Lásd: progalap és bíró... a Bíró csak az stdout-on kapott kimenetet fogja el, rágja meg és próbálja lenyelni, az stderr-t figyelmen kívül hagyja.) Unix világban fontos a megkülönböztetés, mert a legtöbb parancssori eszköz kimenetét valami módon tovább pipeoljuk egy másik feldolgozóba... és ha mondjuk a kimenet közé bekerül egy nem várt hibaüzenet, akkor a későbbi feldolgozás problematikussá válhatott. Szokásos, hogy a rendes kimenetek amiket az elkapó programnak kezelnie kell az megy tovább a pipe-on, a hibaüzenetek meg szorgosan jönnek a terminálra.