Operációs rendszerek 1. – 7. előadás
Külső készülékek kezelése - Biztonság

Soós Sándor

Nyugat-magyarországi Egyetem
Simonyi Károly Műszaki, Faanyagtudományi és
Művészeti Kar
Informatikai és Gazdasági Intézet

E-mail: soossandor@inf.nyme.hu

Tartalomjegyzék.

Tartalomjegyzék

1. Ismétlés

1.1. Emlékeztető az előző órákról

Tárak, tárhierarchia,.

- A tárak hierarchikus rendbe szervezettek:

külső tárak, harmadlagos tárolók
háttértárak, másodlagos tárolók
operatív tár, memória
a processzor regiszterei

- A tárak jellemzői hierarchia szintek szerint:
  - Minél magasabb szinten van egy tároló:
    - annál nagyobb méretű
    - annál lassabb működésű
    - annál nagyobb egységekben címezhető
    - annál hosszabb a tárolási idő
- Alapvető ellentmondás:
  - A különböző tárolási szintek hatékony kezelése a rendszer teljesítményének egyik kulcsa
  - A műveletek elvégzéséhez az adatoknak a processzor regisztereiben kell lenniük. (Miért?)

  - Az összes szükséges adat szinte soha nem fér el a regiszterekben, sokszor a memóriában sem, néha még a háttértárakon sem
  - **A megoldás**: az adatokat rendszeresen mozgatni kell a tárolási szintek között
  - Hogyan?
    * regiszterek $\leftrightarrow$ memória: processzor
    * memória $\leftrightarrow$ háttértár: fájlműveletek
    * háttértár $\leftrightarrow$ külső tárak: felhasználói beavatkozás

- Adatok elérése, címzés a különböző tárolószinteken:
  - **regiszterek**: minden regiszternek külön neve van, bizonyos műveletek csak bizonyos regiszterekkel végezhetők el
  - **memória**: minden memóriarekesz külön-külön címezhető
  - **háttértár**: fájlonként, azon belül rekordonként, blokkonként címezhető
  - **külső tár**: médiánként címezhető, melyik CD/DVD lemezen, szalagon található a keresett adat

- Az adatok mozgatása kétféleképpen történhet:
  1. **Explicit**: (világosan kifejezett) pl. egy utasítással betöltünk egy fájlt a memóriába
  2. **Implicit**: (rejtett, közvetett) a rendszer végzi a háttérben a kényelem fokozása, vagy a hatékonyság növelése érdekében

- A rejtett adatmozgatás tipikus fajtái:
  1. **Virtualizálás**
     - Az alacsonyabb szinten lévő tár címzési módját kiterjesztjük a magasabb szintre
     - Ezzel megnöveljük az alacsonyabb szintű tár méretét (látszólag), de lassabban működik
     - Példa: virtuális memória, lemezen tárolódik, de memória módjára kezeljük, nem fájlként
  2. **Gyorsítótár (cache)**
     - Magasabb szintű elérési módon kezelünk egy alacsonyabb szintű tárat
     - Sokkal gyorsabb
     - De a mérete sokkal kisebb, mint a szimulált tár szokásos mérete
     - Kulcsfontosságú az adatmozgatás szervezése
     - **Lokalitási elv**: ha egy adatra szükség van, akkor nagy valószínűséggel a környezetében lévő adatokra is szükség lesz

  - Ezt használjuk ki a gyorsítótárak adatokkal való feltöltésekor
  - Megfelelő adatcserélési algoritmusokkal és a gyorsítótárak megfelelő méretezésével $80-99\%$-os találati arány is elérhető

- Jellegzetes gyorsítótárak:
  - Processzorba épített hardver-gyorsítótárak (utasítás- és adatcache), a memóriában lévő adatok aktuális részét teszik gyorsabban elérhetővé a processzor számára
  - A memóriában kialakított átmeneti tárterületek (buffer-cache) az éppen használatban lévő fájlok adatai egy részének tárolására
  - Memóriában kialakított virtuális diszk (RAM-diszk, elektronikus diszk)
  - A harmadlagos tárak fájlrendszereit tároló mágneslemez területek

- Mire kell vigyázni a virtuális tárakkal kapcsolatban?
  - Mi történik, ha szabálytalanul állítjuk le az operációs rendszert?
  - A memóriában lévő adatok váratlanul elvesznek
  - A rejtett adatmozgatások félbeszakadnak
  - A háttértárakon lévő adatok inkonzisztens állapotban maradnak
  - A mágneslemezeken lévő adatokat nem tudjuk elérni a hagyományos eszközökkel, ha azok adminisztrációja nem hibátlan

- Hogyan tudunk védekezni az ilyen hibák ellen?
  - Szünetmentes tápegység, akkumulátoros táplálás, notebook
  - **Vigyázat!** Nem csak áramszünet miatt állhat le szabálytalanul az operációs rendszer!
  - Biztonságos szoftvermegoldások (pl. naplózó fájlrendszer, minden végrehajtott műveletet naplóz a rendszer, így rendszerhiba esetén visszaállítható a korábbi állapot

Háttértárak kezelése.

- A memória tartalma addig él, amíg a számítógép működik
- A folyamat szempontjából a memóriában lévő adatok addig élnek, amíg a folyamat fut
- Ha valamilyen adatot meg akar őrizni, akkor háttértárra kell menteni
- A háttértárra írandó adatokat fájlokba kell szervezni
- A felhasználó szempontjából az operációs rendszer legfontosabb feladata a fájlok kezelése (DOS-Disk Operating System)

Fájlkezelés.

- A másodlagos és harmadlagos tárolókon csak fájlokban lehet adatokat tárolni
- A fájlok kezelése az operációs rendszer feladata
- Két szint:
  1. A fájlok, mint tárolási egységek kezelése (egyben)
     - Fájlnév
     - Hierarchikus könyvtárszerkezet
     - Egy, vagy több gyökér (root)
     - Katalógusfájl (directory)
     - Kötetek (volume)
     - Mount
     - A fájl azonosítása: elérési út + fájlnév
  2. A fájlokban lévő adatok kezelése
     - Fájlmodellek
     - Fájlműveletek

Fájlmodellek.

- A fájlban lévő adatok elérésére háromféle fájlmodell használatos:
  1. Soros elérésű (szekvenciális, sequential) fájl
     - mint a mágnesszalag
     - csak sorban lehet írni és olvasni
     - fájlpointer
  2. Közvetlen elérésű (direct) fájl
     - bármelyik adatelem bájt, vagy rekord elérhető a sorszáma alapján
  3. Indexelt, index-szekvenciális elérésű (index sequential access method, ISAM) fájl
     - adatrekordok, adatmezők
     - kulcsmező(k) alapján lehet elérni az adatokat
     - indextábla, indexfájl, rendezett kulcsok, mutató az adatra
     - adatbázis

Fájlműveletek.

1. Megnyitás (open)
2. Lezárás (close)
3. Végrehajtás (execution)
4. Létrehozás (create)
5. Törlés (delete)
6. Adatelérés, írás, olvasás (write, read)
7. Hozzáírás, hozzáfűzés (append)
8. Pozícionálás (seek)

.

Ismétlés vége

2. Külső készülékek és külső kapcsolatok

2.1. A számítógép belseje és a külvilág

A számítógép belseje és a külvilág,.

- Eddig a virtuális gépek olyan objektumaival foglalkoztunk, amelyek a számítógép belsejében találhatók:
  - processzor
  - folyamatok
  - tárak
- Most nézzük, hogyan kapcsolódik a számítógép a külvilághoz!
- Ezeket az eszközöket összefoglaló szóval perifériának nevezzük:
- A határvonal nem éles, pl. a külső tárakat is szoktuk perifériának nevezni

- Nagyon sokféle periféria létezik és ezek száma folyamatosan nő
- Ahhoz, hogy az operációs rendszerek használni tudják ezeket, minden eszközzel nagyobbá és bonyolultabbá kellene válniuk
- Hogy ezt elkerüljük, szabványos módszereket alakítunk ki a perifériák kezelésére
- Két fő típus: (igazából nagyon hasonlítanak egymásra)
  - Fájl
    - Mindent fájlként kezelünk, amibe lehet írni és lehet belőle olvasni
    - Az eszközöket különböző fájlműveletekkel kezeljük: megnyitás, lezárás, írás, olvasás, léptetés, stb.
    - Ilyen eszközök például: képernyő, nyomtató, terminál
    - A fájlműveletek implementálásakor vesszük figyelembe az egyéni részleteket
  - Logikai periféria
    - Absztrakt (virtuális) bemeneti/kimeneti (B/K, I/O) eszköz
    - A programozónak nem kell foglalkoznia azzal, hogy milyen valóságos eszköz van a logikai periféria mögött
    - Ezt nevezzük készülékfüggetlen programozásnak
    - A program futtatásakor az operációs rendszer majd hozzárendel valamilyen fizikai eszközt minden logikai perifériához
    - A fizikai eszköz gyártója ír egy eszközmeghajtó programot (device driver), ami ténylegesen megvalósítja a logikai műveleteket

Hálózati kapcsolatok kezelése,.

- Speciális eset az, amikor a számítógéphez csatlakozó külső eszköz maga is egy számítógép
- Ennek megvalósítására speciális eszközöket építünk a számítógépekbe (hálózati csatoló - network adapter). Ez a fizikai eszköz
- Logikai szinten (szoftveresen) logikai csatlakozókat (socket) hoz létre az operációs rendszer
- Ha összekapcsoljuk két különböző számítógép socketjeit, akkor ezen a kapcsolaton keresztül kommunikálhat egymással a két számítógép
- Ezután egyszerűen fájl típusú perifériaként kezeljük a kapcsolatot
- Egyre közelít egymáshoz a két számítógép és a két folyamat közötti kommunikáció módja, működhetnek üzenetsorok, csővezetékek, különböző pufferek két távoli számítógép között

2.2. Védelem és biztonság

Védelem és biztonság,.

- Amíg egy számítógépet csak egy felhasználó használ és nem kapcsoljuk össze a külvilággal, addig a védelem és a biztonság kérdése nem játszik nagy szerepet
- Amikor egy számítógépet egynél több felhasználó használ, vagy összekapcsoljuk azt a külvilággal, akkor feltétlenül védelmezni kell belülről és kívülről egyaránt
- Definíciók:
  1. Védelem:
     - Védelemnek nevezzük eljárásoknak és módszereknek azon rendszerét, mely lehetőséget teremt a számítógép erőforrásainak programok, folyamatok, illetve felhasználók által történő elérésének szabályozására
     - A rendszer belső objektumaival foglalkozik
  2. Biztonság:
     - A rendszer biztonsága annak a mértéke, hogy mennyire lehetünk bizonyosak a számítógépes rendszer, illetve a rendszerben tárolt adatok sérthetetlenségében
     - A rendszer külvilággal való kapcsolatával foglalkozik

Védelem,.

- Tehát védelemnek fogjuk nevezni a rendszer biztonságának azt a részterületét, ami azzal foglalkozik, hogy a rendszerben legálisan jelenlévő objektumok milyen módon férhetnek hozzá egymáshoz
- Például:
  - Milyen fájlokhoz férhet hozzá egy felhasználó?
  - Milyen memóriaterületet írhat/olvashat egy folyamat?
- Rendszermodell
  - A rendszer különböző objektumok halmaza (hardver-szoftver eszközök, fájlok, processzorok, szemaforok, várakozási sorok, adatszerkezetek, nyomtatók, tárolók, stb.)
  - Az objektumokat valamilyen módon tudjuk azonosítani, névvel, számmal, ...

  - Az objektumok a típusuktól függően különböző műveletekkel rendelkeznek, ezekkel kezelhetők
  - A folyamatok ilyen műveletek sorozatai
  - (Lásd később Objektum-orientált programozás, OOP!)
  - Minden művelethez rendelhetünk jogosultsági előírásokat
  - A műveletet csak az végezheti el, aki rendelkezik a szükséges jogosultságokkal

- Hogyan szervezzük meg a védelmet?
- Ideális helyzet, legbiztonságosabb
  - Minden folyamat minden pillanatban csak azokat a jogosítványokat birtokolja, amelyekre szüksége van a következő művelet végrehajtásához
- Miért jó ez?
  - ⊕ Központilag szabályozzuk a jogosultságokat (odaadjuk-elvesszük)
  - ⊕ Mindig lehet tudni, hogy ki-mit képes megcsinálni
  - ⊕ Könnyen visszakereshető a „tettes”
  - ⊕ Csökkentjük a véletlen hibázás esélyét
- Miért nem jó ez?
  - ⊖ Óriási adminisztratív terhelést okoz
  - ⊖ Gyakorlatban megvalósíthatatlan
  - ⊖ Csak nagyon speciális rendszerekben képzelhető el

Védelmi tartomány,.

- A megoldás: Védelmi tartományok kialakítása
- Védelmi tartomány:
  - Objektumokon végrehajtható műveletekre szóló jogosítványok gyűjteménye
  - Tartalmazhat egy adott objektumon végezhető különböző műveleteket
  - Tartalmazhat különböző objektumokat is rajtuk végezhető műveletekkel
  - Egy-egy objektum-művelet páros több tartományban is szerepelhet
  - A védelmi tartomány lehet statikus, vagy dinamikus

Statikus: egy folyamat futása alatt végig ugyanabban a tartományban marad

Dinamikus: a folyamat tartományt válthat a futása során

- A védelmi tartományokat egy elérési mátrixban definiálhatjuk
- Lássunk egy-egy példát egy statikus és egy dinamikus tartományokat leíró elérési táblázatra!

Elérési mátrix statikus védelmi tartományokkal

		objektumok			
		adat.txt	doc.doc	help.bat	nyomtató
tartományok	A	**olvasás**		**olvasás**	
	B				**nyomtatás**
	C		**olvasás**	**végrehajtás**	
	D	**olvasás, írás**		**olvasás, írás**	

1. ábra. Elérési mátrix statikus védelmi tartományokkal

Elérési mátrix dinamikus védelmi tartományokkal

		objektumok				tartományok			
		adat.txt	doc.doc	help.bat	printer	A	B	C	D
tartományok	A	**olvasás**		**olvasás**			**váltás**		
	B				**nyomtatás**			**váltás**	**váltás**
	C		**olvasás**	**végrehajtás**					
	D	**olvasás írás**		**olvasás írás**		**váltás**			

2. ábra. Elérési mátrix dinamikus védelmi tartományokkal

- A védelmi tartományok használata
  - A folyamat elinduláskor besoroljuk azt valamelyik védelmi tartományba
  - Ez meghatározza, hogy mit tehet meg a folyamat és mit nem
  - Ha dinamikus tartományokat használ a rendszer, akkor a folyamat megteheti, hogy átvált egy másik engedélyezett tartományba

Biztonság,.

- Eddig a védelemmel foglalkoztunk, ami a rendszer belső problémája volt, most áttérünk a rendszer külső védelmére, ezt nevezzük **biztonság**nak
  - *Egy számítógépes rendszer biztonsága, annak a bizonyosságnak mértéke, hogy a rendszer, illetve a rendszerben tárolt információ nem sérülhet meg, vagy illetéktelenül nem kerülhet ki a rendszerből*
- A védelem keretében alapvetően véletlen problémák, programhibák, programok káros mellékhatásai ellen védekezik a rendszer
- Ezzel szemben a biztonság érdekében fel kell készülnünk a szándékos és rosszindulatú támadásokra is
- **Megvalósítható-e tökéletesen biztonságos rendszer?**
- Nincsen abszolút biztonságos rendszer
- Csak relatívan lehet vizsgálni a kérdést
- Legyen „drágább" a támadás, mint az elérhető haszon
- Ha egyszerűen pénzről lenne szó, akkor ez könnyen mérhető lenne, de
  - mennyibe kerül az adat?
  - mennyibe kerül egy adat illetéktelen megváltoztatása?
  - mennyi kárt okoz egy adat illetéktelen megszerzése, és/vagy eltüntetése?
- Ezeket a kérdéseket kell mérlegelni és megválaszolni a biztonsági rendszer megtervezésekor
- Csak úgy lehet elérni a kívánt biztonságot, ha teljes rendszert építünk ki
  - Nem tekinthetünk biztonságosnak egy rendszert, ha informatikailag mindent megtettünk a védelem érdekében, de a nyilvános folyosón áll a szerver, ahol bárki elviheti
- A sérülések oka lehet:
  - véletlen
  - szándékos
    - adatok illetéktelen olvasása
    - adatok illetéktelen módosítása
    - adatok tönkretétele

Felhasználók azonosítása.

- A rendszerhez való jogosulatlan hozzáférés megakadályozásának első lépése a felhasználók azonosítása
- Három módszer:
  1. a felhasználó azonosítása személyes tulajdonságai alapján: ujjlenyomat, retinamintázat, aláírás, stb. (Biometrikus azonosítás)
  2. a felhasználó azonosítása a birtokában lévő tárgyak alapján: kulcs, azonosító kártya, stb.
  3. a felhasználó azonosítása csak általa ismert információ alapján: név, jelszó, esetleg algoritmus
- Az 1. és 2. kategória esetén szükség van speciális perifériákra
- A legelterjedtebb módszer a jelszavas azonosítás

Jelszavas azonosítás.

- Nem igényel speciális eszközt
- Veszélyek:
  - a jelszó megfejtése, kitalálása
  - a jelszó ellopása, lehallgatása
- Védekezés a jelszófejtés ellen:
  - Nehezen kitalálható jelszó választása
  - Megfelelő hosszúságú jelszavak
  - A jelszavak gyakori cseréje
  - Hiba esetén lassítás, letiltás
- Védekezés a jelszólopás ellen
  - Nem jelenik meg a jelszó a képernyőn
  - Csak titkosított csatornán engedünk bejelentkezést
  - A jelszót egyirányú kódolással tároljuk
  - A kódolt jelszavakat is csak a rendszergazda olvashatja
  - Ne írjuk fel a jelszót!
  - Jelszó helyett algoritmikus azonosítás

Külső támadások,.

- Fajtái:
  - Vírusok, férgek
  - Szolgáltatásbénítás (DoS)
  - Betörés (hacker, cracker)
- Férgek:
  - önálló életre képes, és magától terjed
  - nem kell hozzá hordozó program
- Vírusok:
  - hordozó programra van szüksége
  - azt megfertőzve terjed
- Védekezés:
  - Tűzfal
  - Állandóan futó, rendszeresen frissített vírusvédelmi program
- Szolgáltatásbénítás
  - Denial of Service
  - Olyan mennyiségű kérést intézünk a szerverhez, amennyit nem tud kiszolgálni
- Hacker
  - nem ártó szándékú, feltöri a gépet és értesíti a rendszergazdát
- Cracker
  - Rossz szándékú
- Védekezés
  - Tűzfal
  - Biztonsági frissítések rendszeres telepítése
  - Friss alkalmazások használata
  - Körültekintő konfigurálás
    - Csak a szükséges programok fussanak
    - Csak a szükséges dolgokat érjék el a felhasználók

  - A rendszer állandó figyelése, naplózása
  - Gyanú esetén riasztás

- Tűzfal (firewall)
  - Eredetileg olyan házfal, amin nincs ablak, ezért lassabban terjed át rajta a tűz
  - Az informatikában olyan (biztonságos) gép, vagy program, amin a védett gép vagy hálózat forgalma áthalad, és csak azok a csomagok mehetnek rajta keresztül, amelyeket kimondottan megengedünk
  - A gyakorlatban ez leggyakrabban azt jelenti, hogy portszintű szűrést végzünk a forráscím (és a célcím) ill. további feltételek vizsgálata alapján
  - Ma már elengedhetetlen minden számítógépen

3. Befejezés

3.1. Emlékeztető kérdések

Emlékeztető kérdések.

1. Milyen módokon kezelhetjük egységes formában a különböző perifériákat?
2. Hogyan kezeli az operációs rendszer a hálózati kapcsolatokat?
3. Mit nevezünk „Védelem”-nek ebben az összefüggésben?
4. Mit nevezünk „Biztonság”-nak ebben az összefüggésben?
5. Hogyan szervezzük meg a rendszer védelmét?
6. Mit nevezünk védelmi tartománynak? Hogyan lehet definiálni, dokumentálni? Milyen fajtái vannak?
7. Milyen lehetőségek vannak a felhasználók azonosítására?
8. Mire kell figyelni a jelszavas védelem kialakításakor?
9. Milyen külső támadásokra kell felkészülni a rendszer tervezésekor?
10. Hogyan védekezhetünk ezek ellen?

Befejezés.

Köszönöm a figyelmet!