Autorizace

informační systém může poskytovat různé **úrovně ochrany objektů**

1. *žádná ochrana* - postačující pokud dochází k samovolné časové separaci
2. *isolace* - (semi)paralelně běžící procesy jsou zcela odděleny a vůbec o sobě vzájemně neví, systém zajišťuje úplné ukrytí objektů ostatních procesů
3. *sdílení všeho nebo ničeho* - vlastník objektu deklaruje, zda je objekt přístupný všem (public), nebo soukromý (private) a tedy viditelný jen pro něho
4. *sdílení s omezenými přístupy* – testuje se oprávněnost každého pokusu o přístup k danému objektu, k danému objektu a subjektu existuje záznam zda subjekt má právo – konkrétním způsobem - přistupovat k příslušnému subjektu
5. *sdílení podle způsobilosti* (share by capability) - nadstavba předchozího způsobu sdílení, rozsah oprávnění může dynamicky záviset na aktuálním kontextu
6. *limitované použití objektů* - nespecifikuje pouze zda subjekt smí přistupovat k danému objektu, ale i operace, které subjekt smí s objektem provádět

seznam je seřazen podle (implementační) složitosti, které tentokrát přímo úměrně odpovídá kvalita poskytované ochrany

Granularita - kontrola přístupu může být implementována na různých úrovních (byte, věta, soubor, adresář, …), je potřebné volit mezi režií kontroly a dostatečně jemným rozlišením

- paměť
- soubory a data na záznamových zařízeních
- běžící programy
- adresáře souborů
- hardwarová zařízení
- různé datové struktury (stack,…)
- interní tabulky OS
- různé instrukce
- hesla a autentizační mechanismy
- vlastní ochranný mechanismus

cíle ochrany objektů

Kontrolovat každý přístup - subjekt může pozbýt přístupová práva a tedy je nutno mu zabránit v dalším používání objektu

Povolení co nejmenších práv - subjekt by měl mít pouze nejmenší možná oprávnění nutná ke korektnímu plnění jeho úkolu a to i v případě, že případná další práva by

pro něj byla bezcenná - toto uspořádání snižuje možnost průniku v případě selhání části ochranného mechanismu
Ověření přijatelného používání - někdy je daleko podstatnější než přidělení či odepření přístupu moci kontrolovat, co subjekt s daným objektem provádí

Mechanismy ochrany obecných objektů

Adresář (directory)

metodu popíšeme pro případ uživatelů systému v roli subjektů a souborů coby objektů, lze ji však snadno rozšířit na libovolné objekty a subjekty
každý soubor má svého vlastníka, který k němu vlastní veškerá práva včetně práva určovat rozsah oprávnění ostatních uživatelů k tomuto souboru
s každým uživatelem je spojena specielní struktura - *adresář* - obsahující odkazy na všechny soubory, k nimž má daný uživatel nějaké oprávnění, včetně popisu tohoto oprávnění
žádný uživatel nesmí zapisovat do svého adresáře

Nevýhodou může být velký rozsah adresářů a velmi obtížná správa a úpravy takto přidělovaných oprávnění. Rovněž udržení přehledu o tom, kdo k danému souboru má jaká práva může být problematické.

Seznam oprávnění (Access Control List)

opačný přístup k problému
tentokrát je s každým *objektem* udržován seznam informací, které subjekty k němu mají jaká oprávnění
metoda umožňuje snadno přidělovat implicitní práva subjektům případně skupinám subjektů
při vhodném označení subjektů a použití expanzních znaků může být tato metoda dostatečně pružná

Př: Pepek_Group1_Troja
*_Group1_*

seznamy zpravidla bývají udržovány setříděné tak, že záznamy s expanzními znaky jsou na konci - tak stačí hledat první shodu s identifikací subjektu a použít tímto záznamem specifikované oprávnění

Přístupová matice (Access Control Matrix)

řádky matice odpovídají jednotlivým subjektům, sloupce objektům
v políčku daném řádkem a sloupcem je záznam o úrovni oprávnění odpovídajícího subjektu k příslušnému objektu
přístupová matice je zpravidla velmi velká záležitost, zhusta řídká

Způsobilost (Capability)

Způsobilost budeme chápat jako nefalšovatelný token, jehož vlastnictví dává vlastníkovi specifická práva k danému objektu. Lze chápat jako lístek do kina.

jednou z metod zajištění nefalšovatelnosti je, že tokeny se nepředávají přímo subjektům, ale jsou udržovány v chráněné oblasti paměti, přístupné pouze systému

při přístupu k objektu tak systém zkontroluje existenci příslušného tokenu, tento postup lze urychlit tím, že zvlášť udržujeme seznam *Způsobilostí* právě běžícího procesu
výhodou metody je, že dovoluje definovat nové dosud neznámé způsoby používání objektů a přidělovat odpovídající oprávnění
nevýhodou opět poněkud obtížná správa těchto tokenů, zejména odebrání Způsobilosti je netriviální operace

Security Labeling

s každým subjektem a objektem asociujete bezpečnostní label popisující pověření/klasifikaci entity
podpora víceúrovňových modelů

Procedurálně orientovaný přístup

namísto přidělování obecného přístupu k subjektu (čtení, zápis, …) můžeme přidělovat právo používat některých funkcí z rozhraní, prostřednictvím kterého je objekt zpřístupňován
metoda podporuje koncept skrývání a zapouzdřování informací popsaný v minulé lekci
nevýhodou je jistá ztráta efektivity a rychlosti přístupu

Zvládání granularity autorizace

Ochrana po skupinách

uživatelé jsou podle svého zaměření, pracovního zařazení, …, vhodně rozděleny do skupin
pro účely ochrany objektů je svět rozdělen na vlastníka souboru, skupinu, do ktere vlastník patří a ostatní uživatele
předpokládá se, že uživatelé v rámci skupiny potřebují sdílet data
při vytvoření objektu vlastník specifikuje, jaká práva přiděluje sobě, uživatelům ve stejné skupině, ostatním
metoda je jednoduchá, snadno implementovatelná leč neposkytující dostatečně jemné rozlišení, navíc je většinou nutné, aby každý uživatel byl právě v jedné skupině, jinak nastávají problémy s přidělováním práv skupinám

Hesla nebo jiné tokeny

při vytvoření objektu vlastník specifikuje hesla, potřebná pro jisté módy přístupu k objektu, heslo zašle uživatelům, kteří mají mít přístup
systém splní žádost o přístup k objektu pouze tomu, kdo se prokáže odpovídajícím heslem
nevýhodou je, že v případě zapomenutí není možno zjistit, jak heslo vypadalo, v případě, že dojde k vyzrazení hesla je složité nastavit nové, stejně obtížné je odejmout právo přístupu

Dočasné propůjčení oprávnění

mechanismus známý ze systému UNIX.
stejné přidělování práv jako v případě ochrany po skupinách, navíc je možno stanovit, že (spustitelný) soubor smí být prováděn s oprávněním vlastníka
prostřednictvím rutin běžících s oprávněním vlastníka lze řízeně přistupovat k objektům, ke kterým uživatel přímý přístup nemá

problémem popsaných schémat je jistá těžkopádnost, uživatel nemůže selektivně přidělovat práva jistým uživatelům k jistým skupinám objektů
kontrolní matice a podobné metody jsou zase příliš rozsáhlé a obtížně spravovatelné

VAX VMS/SE

ke každému souboru může uživatel vytvořit *Seznam oprávnění* udávající kdo má jaká práva

každý uživatel je členem jedné skupiny, navíc administrátor může vytvořit skupinu typu *obecný identifikátor* , a tuto skupinu mohou uživatelé uvádět v *Seznamech oprávnění*
Seznamy oprávnění mohou být též použity pro přidělování přístupu k ostatním systémovým zdrojům

Systém rolí a skupin

oprávnění jsou sdružována do ucelených souhrnů – tzv. rolí – které odpovídají svým obsahem okruhu práce, kterou vykonává pracovník na určitém zařazení (správce uživatelů, finanční účetní, skladník, ..)
uživatel nezískává oprávnění „po jednom“, ale přidělením role
pro zjednodušení práce bývá k dispozici systém kompozitních rolí, odvozených rolí atd.
namísto práce s jediným uživatelem může být možné definovat a hromadně spravovat celé supinu uživatelů, majících stejná oprávnění

Referenční uživatelé

předpřipravené vzory častých typů uživatelů obsahující např. přiřazené role oprávnění, personalizaci, nastavení ...
- ulehčují správu oprávnění