Информациска безбедност

Предавање:
Авторизација

Проф. д-р Весна Димитрова

Авторизација

- Контролата на пристап е синоним за поимот авторизација.
- Општо, авторизацијата е функција за одредување право на пристап до одредени ресурси кои се од голема важност за безбедност на информациите и компјутерскиот систем.
- Авторизацијата и автентикацијата претставуваат важен дел во безбедноста. Автентикацијата врши верификација на идентитетот на корисникот или процесот, додека авторизацијата одредува дали даден корисник поседува пермиси (дозвола за пристап, привилегии) за да изврши одредена акција.

Авторизација

- Системот за авторизација дава одговори на прашањата:
  -Е овластен пристап до ресурси R?
  -Е овластен за вршење на работата P?
  -Е овластен за вршење на работата P на ресурси R

- Овозможува:
  -Контрола на тоа дали одреден корисник има дозвола да користи одреден ресурс или да изврши одредена акција;
  -Верификација на идентитет на корисник или апликација;
  -Верификација на авторски права врз одредна содржина.

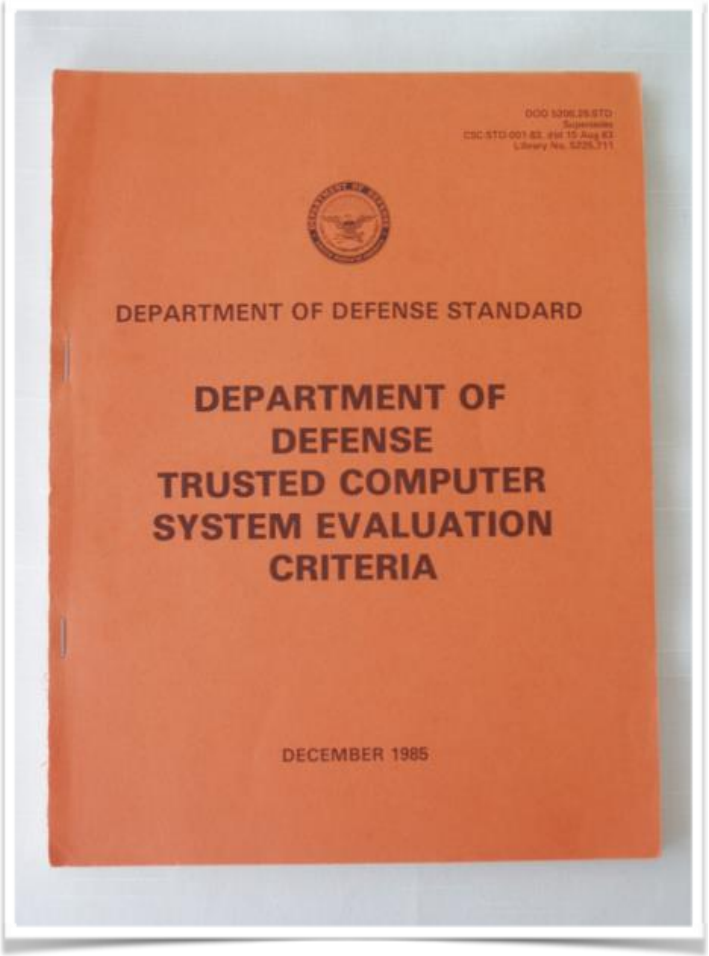
Историја

- Во минатото авторизацијата се сметала за “срцето” на информацискатa безбедност за разлика од денеска каде е составен дел од безбедносниот аспект на системот.

- Во 1983 година е создаден практичен водич за евалуација на системите, сертифицирање и безбедност на повеќе нивоа кој во тоа време служел како патоказ за развивање на потребното ниво на безбедност на информационите системи.

Историја

- Практичниот водич - или т.н. Портокалова книга била создадена од страна на Националната Агенција за Безбедност на САД.
- Иако во тоа време овој прирачник бил од големо значење, заради технолошките револуции неговото значење во денешно време е многу мало.



Историја

- Намерата на оваа книга била да спроведе критериум за евалуација на безбедноста со автоматско процесирање на податоците. Како главни цели на книгата може да се издвојат:
  - Овозможување на критериум спрема кој корисниците ќе можат да го измерат нивото на безбеднот во системите
  - Овозможување на водич за производителите на системи во процесот на подобрување на безбедноста
  - Ја дава основата за безбедносни побарувања.

Историја – Портокалова книга

- Накратко кажано Портокаловата книга овозможува начин со кој се испитува безбедноста на системите и во исто време е водич за тоа како може да се произведат побезбедни продукти.
- Од практична гледна точка Портокаловата книга ги покажува основните побарувања за сертифицирање на систем кој подоцна се користи за евалуација на рејтингот на целокупниот систем.

Историја

- Овој прирачник е поделен во 4 главни делови лабелирани од A до D.
- Секој од овие 4 главни поделби се состои од многу подкласи.
- Четирите главни делови и нивните соодветни класи се следниве:

Историја

- D - Минимална заштита: Овој дел содржи само една класа која е наменета за оние системи кои не можат да ги исполнат побарувањата за некоја повисока класа.

- C - Дискретна заштита: Во овој дел постојат две класи кои овозможуваат дискретна заштита. Тоа значи дека овие две класи не го обврзуваат корисникот за употреба на безбедносни техники, туку овозможуваат техники за лесно откривање на пропусти во безбедноста на системот и тоа:

Историја

- C1 - Дискретна заштита: Во оваа класа системот мора да биде способен да обезбеди глобално лимитирање на пристап до одредени ресурси за индивидуален корисник.
- C2 - Заштита со контрола на пристап: Оваа класа наложува подетална контрола на пристап до ресурси од C1.

Историја

- B - Задолжителна заштита: Овој дел е различен од C во смисла на тоа дека кај C може да се пробие безбедноста, но веројатноста дека обидот за пробивање ќе биде детектиран е голема, додека во B заштитата е задолжителна во смисла дека корисниците не можат да го пробијат системот дури и да пробаат. Класите кои што ги содржи овој дел се следниве:

Историја

- B1 - Лабелирана заштита: Задолжителната контрола на пристап се базира врз основа на специфични лабели т.е. секој податок носи со себе некаков вид на лабела која одредува што му е дозволено на одреден корисник да прави со тој податок.

- B2 - Структуирана заштита: Оваа класа додава заштита од прикриени канали (covert channel protection) и уште неколку технички унапредувања на B1.

Историја

- B3 - Домејни: Како проширување на побарувањата од B2 оваа класа додава побарување дека кодот кој што овозможува безбедност не смее да има можност да биде пристапен и сменет. Значи софтверот кој овозможува безбедност не смее да има пропуст кој што ќе му овозможи на некој корисник да ја смени структурата.

Историја

- A1 - Верифицирана заштита: Овој дел е ист како и B3 во поглед на побарувањата со тоа што овде е додадена и задолжителната употреба на формални методи за докажување на безбедноста на системот.

Историја - The Common Criteria

- Ова бил наследникот на Портокаловата книга и бил интернационален проект спонзориран од владите учеснички во проектот. Овде е поставен стандард за сертифицирање на безбедносни продукти.
- Објавен бил во 2005 година.
- Постојат седум критериуми под кои можат да се сертифицираат продуктите и тоа:

Историја

- EAL1 - Тестирана функционалност
- EAL2 - Структурен тест
- EAL3 - Методолошки тестови и проверки
- EAL4 - Методолошки дизајнирано, тестирано и проверено
- EAL5 - Нецелосно формално дизајнирано и тестирано
- EAL6 - Нецелосно формално верифицирано, дизајнирано и тестирано
- EAL7 - Формално верифициран дизај и тестиран

Историја

- Пример: За да се добие EAL7 рејтинг мора да бидат доставени формални докази за безбедност кои подоцна ќе бидат внимателно анализирани од експертите за безбедност.
- Најголем број продукти се сертифицирани со EAL4, бидејќи тоа претставува минимумот кој што е потребен.

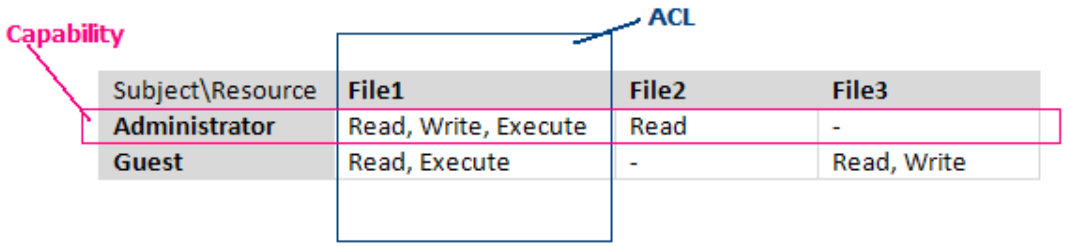
Матрица за пристап

- Класичниот поглед на авторизација започнува со матрицата за пристап на Лампсон. Оваа матрица ги содржи сите важни информации кои му се потребни на оперативниот систем да може да препознае кој корисник какви привилегии има.

	Object 1	Object 2	Object 3	Object 4
Domain 1	{read, write}	{execute}		
Domain 2		{write}		{print}
Domain 3	{execute}		{read}	{print}

Матрица за контрола на пристап

- Да дефинираме субјект како корисник на системот и објект како ресурс на системот. Фундаменталните функции на авторизацијата се access control lists или ACLs и capabilities lists или C - листи. И ACLs и C - листите се наследени од матрицата за пристап на Лампсон каде што секој ред е субјектот, а колоната објектот. Според тоа пристапот на одреден субјект S е дозволен/недозволен/парцијално дозволен врз одред



Subject\Resource	File1	File2	File3
Administrator	Read, Write, Execute	Read	-
Guest	Read, Execute	-	Read, Write

Матрица за контрола на пристап

- S-множество субјекти, кое се состои од сите активни ентитети (корисници, процеси).
- O-множество објекти, ентитети кои треба да бидат заштитени (процеси, датотеки...).

	OS	Accounting Program	Accounting Data	Insurance Data	Payroll Data
Bob	rx	rx	r	—	—
Alice	rx	rx	r	rw	rw
Sam	rwx	rwx	r	rw	rw
acct. program	rx	rx	rw	rw	r

ACLs и C - листи

- ACLs и C - листите се користат при авторизација на корисниците. Проблемот со овие листи настанува кога имаме илјадници корисници и објекти кои што би направиле матрица со милиони пермисии што би било голем товар на системот.
- За да се задржат перформансите на системот во вакви случаи овие матрици се делат на т.н. делови кои што можат да се менаџираат. Постојат два начина на кои што може да се одделат овие матрици и тоа:

ACLs и C - листи

- Прв начин е да се подели матрицата на колони и да се зачува секоја колона за одреден објект и при секој обид за пристап се гледа само групата на корисници за тој објект.
- Ваквата поделба по колони се нарекува листа за контрола на пристап.

(Bob, —), (Alice, `rw`), (Sam, `rw`), (accounting program, `rw`).

ACLs и C - листи

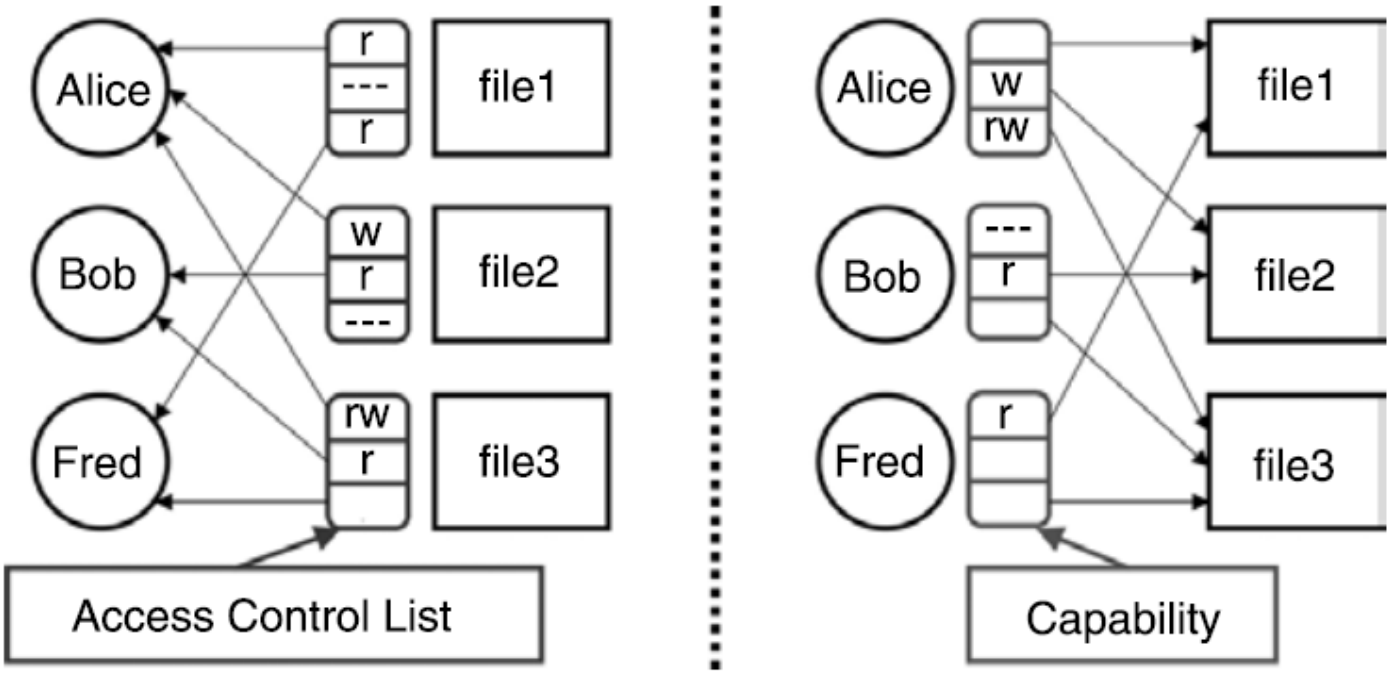
- Вториот начин е да се подели матрицата по редови каде што секој ред се чува со својот субјект. Притоа кога некој субјект пробува да изврши некоја операција, тогаш од редот се гледа дали операцијата е дозволена за тој субјект.
- Овој листа се нарекува листа на можности или C - листа.

$$(\text{OS}, \texttt{rx}), (\text{accounting program}, \texttt{rx}), (\text{accounting data}, \texttt{r}),$$
$$(\text{insurance data}, \texttt{rw}), (\text{payroll data}, \texttt{rw}).$$

ACLs и C - листите

- На прв поглед изгледа дека овие две листи се идентични, но во суштина тие овозможуваат два различни начини за зачувување на истата информација.
- Во реалноста C - листите се повеќе популарни од листите за контрола на пристап.
- Пример: Кај C-листите врската помеѓу корисници и податоци е вградена во самиот систем, додека кај листите за контрола на пристап потребен е посебен метод за врска помеѓу корисници и податоци.

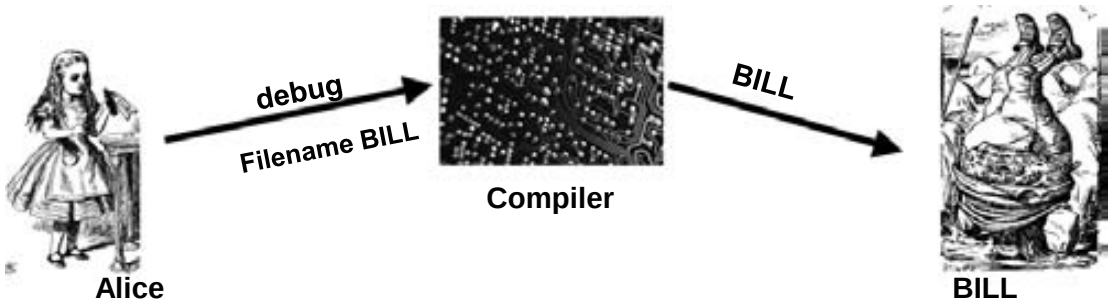
ACLs vs. C-list



Насоката на стрелките е различна, кај ACL имаме насока од ресурсите кон корисниците, додека кај Capability насоката е обратна.

Confused Deputy- “збунет помошник”

	Compiler	BILL
Alice	x	—
Compiler	rx	rw



Confused Deputy- “збунет помошник”

- Кога Alice се обидува да пристапи до датотеката BILL оди преку компајлерот кој дејствува врз основа на сопствените привилегии наместо оние на Alice.
- Кај ACL овој проблем е тешко да се избегне, додека кај Capabilities релативно лесно се надминува.
- Кај систем базиран на Capabilities кога Alice ќе го повика компајлерот ја предава и C-листата и понатаму компајлерот дејствува соодветно со привилегиите на Alice.

Модели за безбедност на повеќе нивоа

- Овие модели служат за да ни покажат што треба да се заштити во системот, но не одговараат на прашањето како да се спроведе оваа заштита. Значи овие модели се само показатели за тоа што треба да се заштити, а не функции кои спроведуваат заштита.

- И во овие модели земаме субјект да е корисникот, а објект да е податокот. Според тоа клацификацијата се става на објектите, а ниво на дозволата за пристап на субјектите.

Модели за безбедност на повеќе нивоа

- Пример: Одделот за одбрана на САД има 4 нивоа на класификација и дозвола за пристап и тоа:
- TOP SECRET>SECRET> CONFIDENTIAL> UNCLASSIFIED
- Според тоа субјект со дозвола за пристап SECRET има дозволен пристап до објекти од ниво SECRET или помало, но нема пристап до TOP SECRET нивото.

Модели за безбедност на повеќе нивоа

- Овој модел на безбедност на повеќе нивоа е потребен кога субјекти и објекти на различни нивоа користат исти ресурси. Целта е да се засили контролата на пристап со тоа што ќе се ограничат субјектите да пристапуваат до ресурси на ниво за кое имаат дозвола.

- Во денешно време овој модел на безбедност на повеќе нивоа се употребува насекаде. На пример, голем број од компаниите го употребуваат за класификација на документи кои не би требало да бидат достапни за вработени од понизок ранг.

Модели за безбедност на повеќе нивоа

- Истотака постои голем интерес за овој модел во апликации кои служат како мрежни firewalls. Целта во овој случај е да се држи натрапникот на што пониско ниво со што би се намалила штетата која што би можела да биде предизвикана доколку биде заобиколен firewall.
- Постојат и други модели кои што се пореални, но во исто време се и покомплексни и потешки за анализа и верификација.

Bell-La Padula Model

- Bell-La Padula моделот е модел за контрола на пристап и е доста користен во владата на САД. Попознат е како “no read up, no write down” модел. Тој користи множество од правила за контрола на пристап што користат лабели на објектите и дозвола за пристап на субјектите. Овие лабели варираат од Unclassified, For Official Use Only, Confidential, Secret, Top Secret, итн.

- Овој модел дава големо значење на тајноста на податоците и контролираниот пристап до овие тајни податоци. При обид за пристап од страна на субјект се споредува нивото на субјектот со нивото на објектот до кој има намера да пристапи и со тоа или ќе му се додели пристап на субјектот или ќе се одбие.

Bell-La Padula Model

- Кај овој модел дефинирани се следниве правила за контрола на пристап:
  - Субјект на одредено безбедносно ниво не смее да чита објект на повисоко безбедносно ниво.
  - Субјект на одредено безбедносно ниво не смее да запишува објекти на помало безбедносно ниво.

Bell-La Padula Model

- Исто така постои и карактеристика која што не дозволува субјектите да запишуваат објекти во повисоко безбедносно ниво.
- Ова е додадено ниво на интегритет.

Biba Model

- Безбедносниот Биба модел е развиен за да се покажат слабостите на Bell-La Padula моделот. Биба моделот се фокусира на интегритетот што всушност е делот кој недостасува во Bell-La Padula моделот кој е фокусиран на тајноста. Исто како и кај Bell-La Padula, овој модел користи субјекти и објекти. Разликата во Биба моделот е тоа што објектите и субјектите се групирани во ниво на интегритет наместо во лабелирани безбедносни нивоа. Овој модел се карактеризира со фразата “no read down, no write up”.

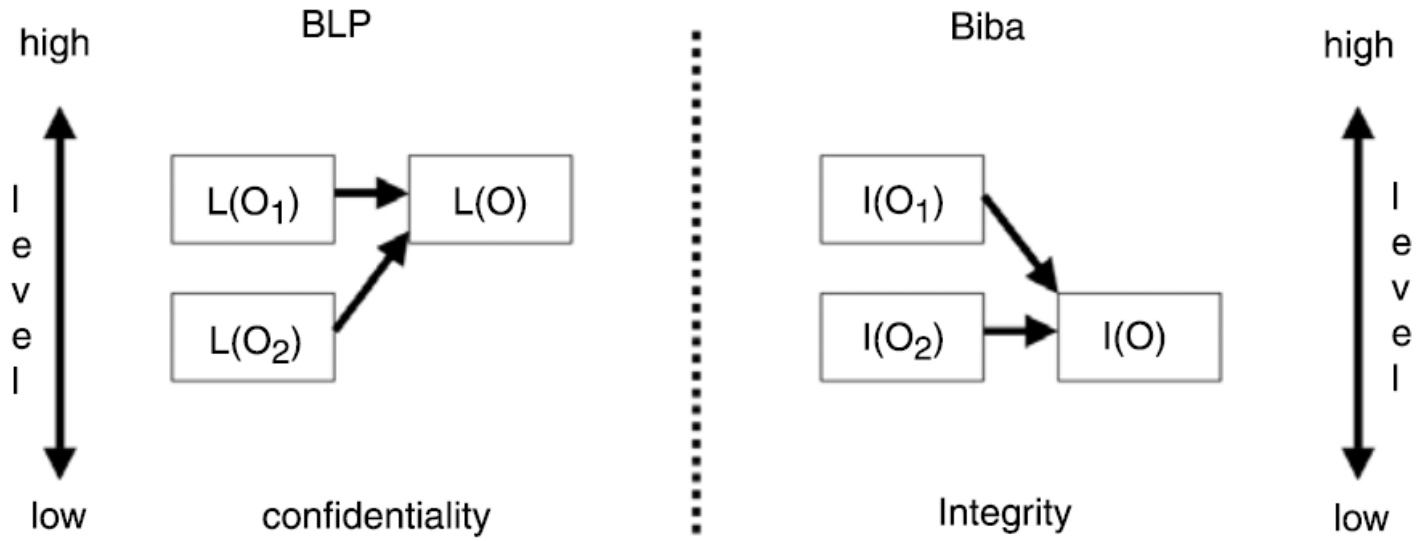
Biba Model

- За да се зачува интегритетот, субјектите можат да креираат содржина на нивното ниво на интегритет или под нивното ниво на интегритет, а да гледаат содржина од нивното или погорно ниво на интегритет.
- Ова помага во спречување на оштетување на податоци и во исто време го задржува интегритетот.

Biba Model

- Безбедносни правила кај Биба моделот:
  - Субјект на дадено ниво на интегритет не смее да чита објект од пониско ниво (no read down).
  - Субјект на дадено ниво на интегритет не смее да запишува објекти на повисоко ниво на интегритет (no write up).

Bell-LaPadula vs. Biba's Model



Повеќекратна безбедност

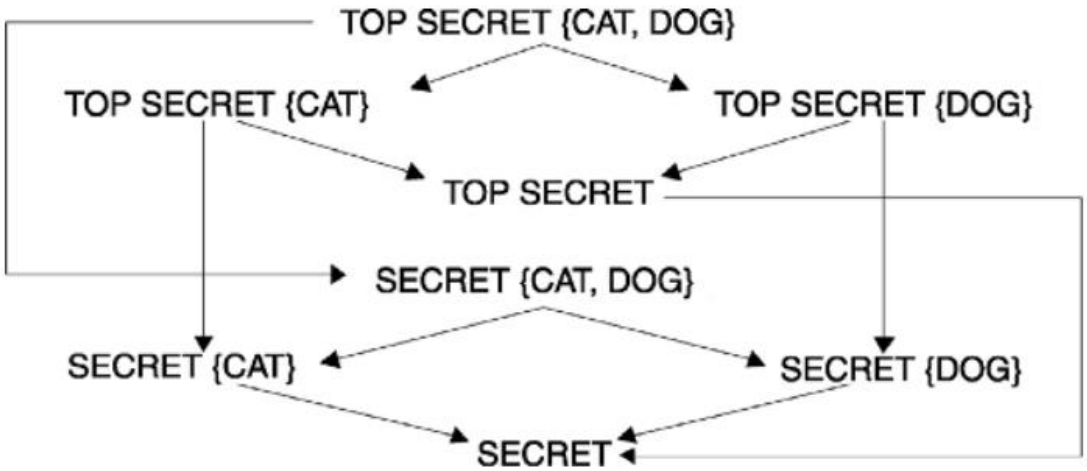
- Се користат секции/оддели за поголема заштита и подобро спречување на информациите што поминуваат низ безбедносните слоеви
- SECURITY LEVEL {COMPARTMENT}
  - TOP SECRET {CAT,DOG}
  - Ниво - TOP SECRET
  - Секции - {CAT} {DOG}

Повеќекратна безбедност

- Секциите се користат според принципот “потреба да се знае” информацијата.
- На корисниците пристапот им е дозволен само ако имаат потреба од истата.
- Ако не е потребно да се знаат сите информации што се наоѓаат на TOP SECURITY нивото се користат секции за да се лимитираат непотребните.

Повеќекратна безбедност

- TOP SECRET {CAT} дозволата нема пристап во TOP SECRET {DOG}. Има пристап во SECRET {CAT}, но не и во SECRET {CAT, DOG}.



Прикриен канал (Covert Channel)

- Covert Channel е тип на напад кој што создава можност за трансфер на објекти помеѓу субјекти или процеси кои што не би смееле да комуницираат според дефинираната безбедносна полиса.
- Се нарекува Covert Channel, бидејќи е скриен од механизмите за контрола на пристап со тоа што не користи легитимни механизми за пренос на податоци како на пример read и write и според тоа не може да биде детектиран или контролиран од страна на безбедносните механизми.

Прикриени канали

- Скоро е виртуелно невозможно да се елиминираат Covert Channels, па фокусот е ставен на тоа да се лимитира капацитетот на ваквите канали.
- Моделите на безбедност на повеќе нивоа се дизајнирани да ги забранат легитимните канали за комуникација, но Covert Channels не ги користат овие легитимни канали/механизми за пренос на информации.
- Пример како се пренесуваат ресурси помеѓу корисници на различни нивоа кои имаат пристап до заеднички податок и со тоа ја нарушуваат безбедноста во моделот за безбедност на повеќе нивоа:

Прикриени канали

- Да претпоставиме дека Корисник А има TOP SECRET привилегија и Корисник В има CONFIDENTIAL привилегија. Ако ресурсите се поделени за сите корисници, тогаш Корисник А и Корисник В можат да се договорат дека ако Корисник А сака да му испрати 1 на Корисник В, тогаш ќе креира Датотека 1, а ако сака да му испрати 0 нема да креира никаква датотека. Тогаш Корисник В ќе провери дали Датотека1 постои и ако постои ќе знае дека Корисник А му испратил 1 доколку Датотека 1 не постои, тогаш Корисник А му испратил 0.

Прикриени канали

- Во овој случај е поминат еден бит (1 или 0) преку Covert Channel. Според безбедносната полиса Корисник B не може да ја отвори Датотека 1, бидејќи нема привилегии за другото ниво, но да речеме дека може само да излиста и да види дали датотеката постои.
- Протекување на 1 бит не е загрижувачки, но на овој начин Корисник A може на пренесе голем број на информации.

Прикриени канали

- Да претпоставиме дека Корисник А и Корисник В се договорени да Корисник В проверува на секоја минута дали Датотека 1 постои. Според нивниот претходен договор ако Датотека 1 постои тогаш Корисник А му праќа 1 на Корисник В во спротивно му праќа 0. На овој начин Корисник А полека, но сепак му праќа на Корисник В TOP SECRET информација.

Прикриени канали

- Covert Channels се насекаде. На пример, редот на документите за печатење може да биде користен како сигнал за некоја информација слично како примерот претходно.
- Потребни се три работи за да постои Covert Channel.
  - Прво испраќачот и примачот треба да имаат право на пристап до поделен ресурс.
  - Второ испраќачот мора да е во можност да менува нешто на/или податокот.
  - Трето испраќачот и примачот мора да имаат синхронизирана комуникација.

Прикриени канали

- За да се елиминира Covert Channel би требало сите споделени ресурси помеѓу корисниците да бидат елиминирани. Повеќе од очигледно е дека ваков систем не би бил од корист…
- Да го земеме TCP протоколот како еден современ пример за Covert Channel. TCP header-от има т.н. резервирано поле за употреба, а всушност не се употребува за ништо. Според тоа ова поле може многу лесно да се употреби за да се пренесе некоја информација.

CAPTCHA

- Туринговиот тест бил претставен во 1950 година од страна на Алан Туринг. Комплетно автоматизиран, јавно достапен Турингов тест за разликување на компјутер од човек или CAPTCHA е тест што човек може да го помине меѓутоа компјутер не. Ова може да се смета како инверзен Турингов тест.
- Со други зборови CAPTCHA е програм кој што може да генерира и оценува тестови кои самиот тој неможе да ги реши.

CAPTCHA

- Бидејќи CAPTCHA се дизајнирани за да спречат пристап до ресурси за сите освен луѓето, можеме да го гледаме како еден вид на механизам за контрола на пристап.
- Барањето за изработка на овој систем било да е лесно за луѓето да го погодат, но во исто време да не невозможно компјутер да го реши дури и компјутерот да има пристап до CAPTCHA софтверот.

CAPTCHA

- Од перспектива на напаѓачот, единствено непознато нешто е случајноста што е употребена за генерирање на специфични CAPTCHA. Исто така се препорачува да се има различни типови на CAPTCHA во случај некоја личност да неможе да помине одреден тип.
- CAPTCHA се употребува на голем број од веб сајтови.

CAPTCHA

- тест познат и како “златно правило ” во вештачката интелигенција
- тестот го генерира и гради одредена компјутерска програма
- компјутерот има пристап до изворниот код којшто се користи за генерирање на тестот
- не постои компјутер којшто може да го помине тестот

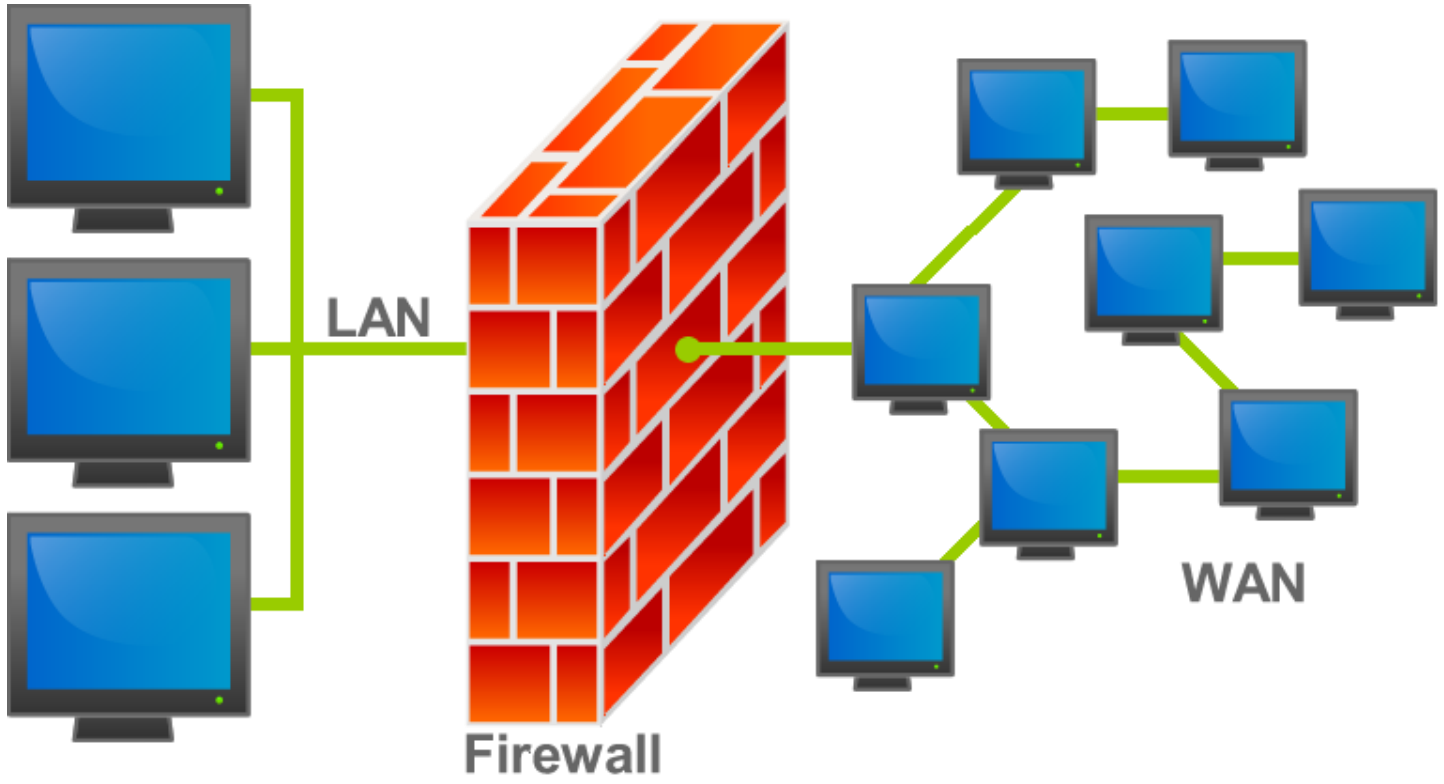
CAPTCHA

- Се користи кај бесплатните e-mail сервиси (Yahoo, Gmail…),...

CAPTCHA


509876
2c29d
0c8179
afootaside
>BEV
fcfa8da
i92wvd
42429
couldnt47
BsDnD5
xybin
DPZPQV
q24vp
Ht3s1F
2c488
LHJSGS
8A2C9

Firewalls (Огнени Ѕидови)



Firewalls

- Мрежниот Firewall (Огнен Ѕид) е сместен помеѓу внатрешната мрежа (локална мрежа) и надворешната мрежа (Интернет)
- Задачата на огнениот ѕид е да се утврди што да се дозволи во и надвор од внатрешната мрежа (локална мрежа)

Firewalls

- Класификација на firewall:
  - *Packet filter* е firewall којшто работи на мрежното ниво
  - *Stateful packet filter* е firewall којшто работи на транспортното ниво
  - *Application proxy* е firewall којшто работи на апликативно ниво и функционира како proxy

Packet Filter

- Испитува (филтрира) пакети до мрежниот слој

  - изворната IP адреса, изворната порта, дестинацискатa порта и TCP знаменцата (SYN, ACK, RTS)

  - може да филтрира врз основа на влезот или излезот (може да има различни правила за филтрирање за дојдовни и појдовни пакети)

Packet Filter

- **Предности:**
  - ефикасност, брзина
  - брза обработка на податоците до мрежно ниво

- **Недостатоци:**
  - нема концепт за состојба, така што секој пакет се третира независно од сите други
  - не може да ја испита TCP конекцијата
  - слеп за апликациско ниво каде се сместени малициозните програми

Packet Filter

- Пакет филтрите се конфигурирани со листи за контрола на пристап кои се различни од претходно споменатите. Овие листи служат за да се забранат пакети од одредени сервиси, на одредени порти итн.

- Заобиколувањето на овој firewall е многу едноставно од причина што не се води сметка за состојба на конекцијата, па корисникот може да ги пронајде отворените порти кои подоцна ќе ги искористи за напад.

Stateful Packet Filter

- Овој тип на firewall е како и претходниот со тоа што кај овој тип се додава и состојбата

  - се испитува состојбата на TCP и UDP конекциите

  - оперира на транспортно ниво (таму се одржуваат информации за врските)

Stateful Packet Filter

- **Предности:**

  - ги следи тековните врски

  - спречува многу напади како на пример ACK scan (скенирање на отворени порти)

- **Недостатоци:**

  - не може да ги испита податоците од апликацијата

  - побавен е од Packet Filter

  - потребна му е поголема обработка

Aplication proxy

- Application Proxy firewall ги процесира сите пакети кои доаѓаат се до апликациско ниво.

  - способен е да верифицира дали пакетите се легитимни и дали податоците во пакетите се безбедни.

- **Предности:**

  - комплетна слика на конекциите и на апликациските податоци (сеопфатен поглед)

- **Недостатоци:**

  - брзина

Personal Firewall

- Персоналните firewall се користат за заштита на одреден хост или мали мрежи како што се домашните мрежи.
- Било кој од претходно споменатите firewalls може да се користат за оваа намена.

Детекција на напади (Intrusion detection)

- Примарниот фокус на компјутерската безбедност е превенција од натрапници (intrusion prevention), каде целта е да се држат натрапниците надвор од нашите системи или мрежи.
- Автентикацијата претставува еден вид начин за заштита од напади, исто како што се и повеќето видови на заштита од вируси.

Детекција на напади

- Што да направиме кога превенцијата од напади е неуспешна?
  - Системи за откривање на напади (Intrusion detection systems) – IDSs
  - Целта на овие системи е да се откријат нападите пред, за време или после нивното случување.
  - Детекција на напади е моментално многу активно истражувана тема.

Детекција на напади

- Кои се напаѓачите што IDS системите се обидуваат да ги откријат?
- Какви видови на напади би можел еден натрапник да започне?

Архитектури на IDS

- Постојат две основни архитектури за Системите за откривање на напади:
  - Хост базирани IDS, чиј што метод за детекција се базира на пратење на активноста на хостовите.
  - Мрежно базирани IDS, чиј метод се базира на детекција на промена во мрежниот сообраќај. Овие системи се дизајнирани за детектираат напади како denial of service, скенирање на порти итн.

Методи за откривање на напади

- Signature – based IDSs
  - Се обидуваат да откријат напади врз основа на „потписи“.
- Anomaly – based IDSs
  - Се обидуваат да дефинираат нормално однесување на системот и да обезбедат предупредување кога во системот се работи нешто надвор од нормалното.

Signature – based IDSs

- Предности на signature-based IDSs
  - Едноставнот, ефикасност, одлична способност за откривање на познати напади.
- Недостатоци на signature-based IDSs
  - Бројот на потписи може да стане многу голем со што се намалува ефкасноста.
  - Системот може да ги детектира само познатите напади.
  - Постои веројатност системот да ги пропушти дури и најмалите варијации на добро познатите напади.

Anomaly – based IDSs

- Овие системи се базираат на откривање на некое необично однесување на системот.
- Постојат неколку главни предизвици својствени за ваквиот систем
  - мора да се одреди што би се сметало за нормално однесување.
  - самата дефиниција за нормално однесување мора да се адаптира и да еволуира според промените во системот затоа што во спротивно бројот на грешни аларми ќе се зголеми.

Anomaly – based IDSs

- мора да се знае точно колку променетото однесување на системот се разликува од нормалното.
- Статистиката е очигледно неопходна при развој на ваков IDS базиран на аномалии.

- Како може да се измери нормално однесување на системот?
  - Без разлика кои карактеристики ќе се мерат, мерењата мора да бидат земени во времето кога тоа однесување на системот се случува.
  - Секако овие мерки не треба да се прават за време на напад на системот, бидејќи во тој случај нападот би се сметал за нормално однесување.

Anomaly – based IDSs

- Пример за техники кои се користат за да се распознае вообичаено од невообичаено однесување:
  - Баесови анализи
  - linear discriminant анализа LDA
  - quadric discriminant анализа QDA
  - невронски мрежи
  - скриени маркови модели
- Понекогаш се користат и модели од вештачка интелегенција

Пример за anomaly-based IDSs

- Претпоставуваме дека со текот на времето Alice пристапува до четири датотеки: F_0 , F_1 , F_2 , F_3 со честота:

H_0	H_1	H_2	H_3
.10	.40	.40	.10

Сега претпоставуваме дека во некој нов временски интервал Alice има пристапено до датотеката F_i со честота A_i , за i=0,1,2,3

A_0	A_1	A_2	A_3
.10	.40	.30	.20

Пример за anomaly-based IDSs

- Дали последните пристапувања на Alice кон датотеките се во границите на нормалното однесување?
- Ја користиме следнава статистика:

$$S = (H_0 - A_0)^2 + (H_1 - A_1)^2 + (H_2 - A_2)^2 + (H_3 - A_3)^2$$

За да одговориме на поставеното прашање, дефинираме:

$S < 0.1$ како нормално однесување.

Пресметуваме:

$$S = (0.1 - 0.1)^2 + (0.4 - 0.4)^2 + (0.4 - 0.3)^2 + (0.1 - 0.2)^2 = 0.02$$

Заклучок ?

Пример за anomaly-based IDSs

- Пристапувањата на Alice кон датотеките може да варираат со текот на времето и ова би требало да се предвиди во IDS системите.
- Тоа го правиме со ажурирање на долгорочна историја на вредности H_i според формулата:

  $H_i = 0.2 * A_i + 0.8 * H_i$ *за* $i= 0,1,2,3$ и добиваме:

H_0	H_1	H_2	H_3
.10	.40	.38	.12

- Ако имаме нови вредности на Ai (0.1, 0.3, 0.3, 0.3) добиваме:

$S = (0.1 - 0.1)^2 + (0.4 - 0.3)^2 + (0.38 - 0.3)^2 + (0.12 - 0.3)^2 = 0.0488$

Бидејќи $S = 0.0488 < 0.1$, повторно заклучуваме дека однесувањето е нормално.

Задача

- Дадени се следниве табели:

H_0	H_1	H_2	H_3
.10	.38	.364	.156

A_0	A_1	A_2	A_3
.05	.25	.25	.45

Пресметајте дали е ова нормално за Alice.