СИСТЕМСКИ СОФТВЕР ИСПИТ

1. Што представува системски софтвер?

-Системскииот софтвер го сочинуваат програми што се неопходни за работа на компјутерот и кои ја управуваат и контролираат целокупната работа на хардверот, т.е. ги опфаќа т.н. процесорски програми кои воспоставуваат врска помеѓу хардверот и апликативниот софтвер и ја координира работата на хардверот во склад со барањата на апликативниот софтвер, обезбедувајќи максимално ажурирање на севкупните компјутерски ресурси.

1. Од што се состои основниот системски софтвер?

-Основниот системски софтвер се состои од неколку видови програми:

-Иницијален полнач (bootstrap loader)

-Дијагностички рутини

-Основен влезно-излезен систем

-Драјвери

1. Што претставуваат контролни, а што услужни програми?

-Контролните програми служат за управување со хардверот (нивната задача е да го тестираат, да го контролираат хардверот и да управуваат со него во текот на извршувањето на сите влезно-излезни операции), додека услужните програми се користат за оперирање со компјутерскиот систем, како негово конфигурирање, поставување на работни параметри, групирање на програмите и сл.

1. Историјат

-Компјутери од првата генерација (1945 – 1955) – со вакуумски цевки, биле со огромни димензии и многу скапи. Најчесто ги користела војската. Човекот имал потполна контрола над компјутерскиот систем.

-Компјутери од втората генерација (1955 – 1965) – со транзистори, биле помали, подоверливи и поевтини. Освен војската ги купувале и големите корпорации и универзитети. Се пишувале програми на програмскиот јазик FORTRAN.

-Компјутери од третата генерација (1965 – 1980) – се прават со интегрални кола, се прават два вида на компјутери: една брза верзија (како IBM 7094) и една послаба (како IBM 1401), што претставува скап потфат.

-Четврта генерација компјутери (1980 – 1990) – прв пат се појавуваат персоналните компјутери, развојот на персоналните компјутери започна со појавата на LSI чиповите (чиипови со висок степен на интеграција), покрај класичните оперативни системи се јавуваат и мрежно оперативните системи и дистрибуираните оперативни системи.

1. Што претставуваат мрежни системи, а што дистрибуирани?

-Мрежно оперативните системи ги карактеризираат компјутерите поврзани во мрежа (овие компјутери содржат релативно висок степен на автономија – секој компјутер има свој оперативен систем, и се во можност меѓусебно да разменуваат податоци со помош на одредени протоколи). Дистрибуираните оперативни системи се многу посериозна варијанта на мрежното опкружување. Освен споделување и миграција на датотеки овозможува и споделување на процеси т.е. програми.Тука, постојат повеќе компјутери поврзани во мрежа, но само еден оперативен систем управува со сите ресурси во мрежата.

1. Која е разликата помеѓу нив ? ^

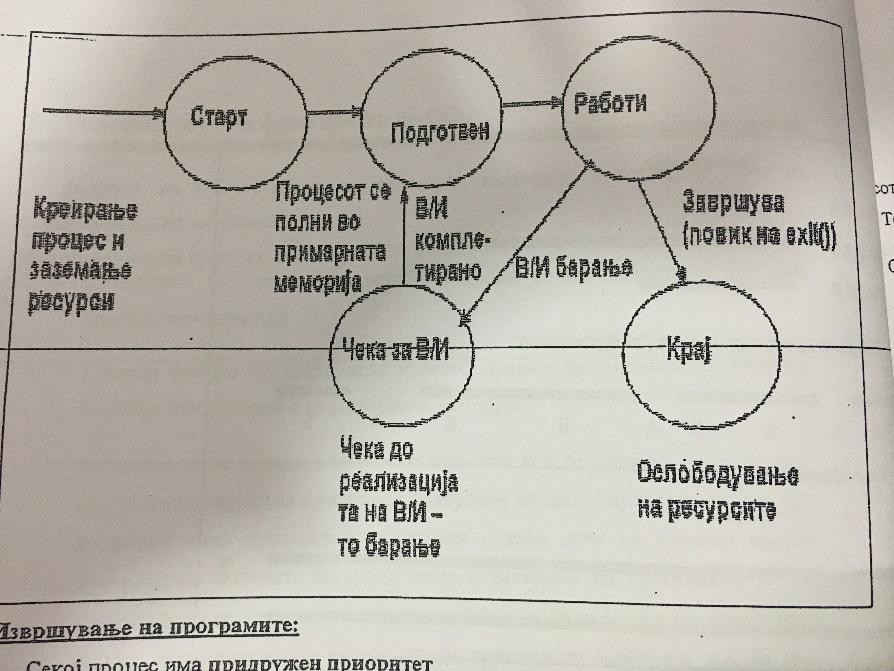
-Во мрежно оперативните системи корисник на еден компјутер може да се пријави на друг, да преземе некои датотеки и сл., додека кај дистрибуираните освен споделување и миграција на датотеки овозможено е и споделување на процеси т.е. програми. Мрежните оперативни системи имаат голем степен на автономија, додека дистрибуираниот оперативен систем се однесува како една целина.

1. Основни функции на ОС

-Распределба на процесорското време

-Распределба на компјутерската меморија

-Организација на пристапот до надворешните мемории

-Доделување на периферни уреди 7.Како изгледа животниот век на

процесот

8.Кои се причините за да се креира процес?

-Корисникот се најавува

-Корисникот стартува програма

-Оперативниот систем креира процес за да обезбеди некоја услуга

-Некои програми стартуваат други процеси 9.При активирање на кои настани се креира процес?

-Иницијализација на системот

-Извршување на повик на системот за креирање на процес

-Кориснички повик да се креира процес

-Иницијализација на процес за пакетна обработка 10.Како може да заврши еден процес?

-Нормален крај

-Постоење на грешла

-Фатална грешка

-Уништен од страна на друг процес 11.Што представува податочна патека?

-Податочна патека е оној дел од процесорот кој ги содржи аритметичко-логичката единица, нејзините влезови и излези. Се состои од 32 регистри, ALU и магистрали.

1. Што претставува контролна единица?

-Контролната единица го надгледува извршувањето на сите инструкции и преносот на сите информации. CU зема инструкции од меморија, ги декодира, обезбедува да има податоци на вистинско место во вистинско време, и кажува на ALU кои регистри да ги користи, ги сервисира интераптите.

1. Што претставуваат регистрите, какви видови постојат и која е нивната функција?

-Регистар претставува хардверски уред за чување на бинарни податоци, се користат за чување на адреси, контролна информација или податоци за извршување на програмата. Постојат повеќе специјализирани регистри:

-за чување информации, за шифтирање

-споредување вредности,броење

-scratchpad регистри за привремено чување информации

-индексни регистри за контрола на програмски јамки

-stack pointer регистри за работа со стекови со информација за

процесите

-статус регистри кои чуваат битови за overflow, carry, zero

-регистри за општа намена достапни на програмерот

1. Како работи контролната единица?

- CU зема инструкции од меморија, ги декодира, обезбедува да има податоци на вистинско место во вистинско време, и кажува на ALU кои регистри да ги користи, ги сервисира интераптите.

1. Што представува собирница, од што се состои, какви видови постојат?

-Собирници претставуваат множество линии низ кои битовите се движат паралелно и му помага на CPU да комуницира со другите компоненти. Типичната собирница се состои од податочни линии, адресни линии, контролни линии и линии за напојување. Собирниците се делат според типот на информација која ја пренесуваат и уредите кои ја користат:

-Процесор-меморија собирниците се кратки, брзи и прилагодени да го зголемат преносот на податоци од меморија

-I/O собирниците се подолги и дозволуваат приклучување на различни периферни уреди.

-Backplane собирницата е вградена во куќиштето и ги поврзува процесорот, I/O уредите и меморијата

1. Како се врши резервирање на собирници кај системи со повеќе од еден master bus уред?

-Кај системи со повеќе од еден master bus уред потребна е арбитражна шема за доделување на приоритет на master bus уредите за користење на собирницата.

1. Арбитражни шеми

-Daisy chain arbitration – преку контролната линија се пренесува дозвола за користење на собирницата од направа со највисок приоритет до направа со најнизок.

-Централизирана паралелна арбитража – секоја направа испраќа барања преку своја контролна линија за користење на собирницата, а централен арбитер одлучува кој ќе ја користи собирницата

-Дистрибуирана арбитража – слична шема на претходната но уредите сами одлучуваат кој треба да ја добие собирницата.

-Дистрибуирана арбитража со откривање на судири – секој уред испраќа барање. Ако собирницата открие повеќе истовремени барања уредите мора да испратат други барања за користење на собирницата.

1. Како се врши размена на информации кај влезно излезните уреди преку влезно излезни регистри?

-CPU разменува информации со В/И уреди преку В/И регистри.

-Мемориски мапирани В/И – В/И регистрите се јавуваат во мемориската мапа на КС и нема разлика меѓу пристапување до меморија и пристапување до В/И уред. Пристапот е побрз но се троши меморија.

-Инструкциски базирани В/И – CPU користи специјализирани инструкции за изведување на читање или запис од влезни и излезни уреди. Не се троши меморија но се потребни специјализирани В/И инструкции.

1. Што претставуваат прекини и од која причина можат да се појават да се објасни

-Интерапти се настани кои го менуваат нормалниот тек на извршување на програмата.

-Интераптите може да се предизвикани од различни причини:

-В/И барања

-Аритметички грешки(делење со нула)

-Underflow/Overflow

-Хардверски грешки (мемориска грешка во парност)

-Кориснички дефинирани прекини (break points при debugging на

програми)

-Page fault (мемориска страна не е вчитана)

-Грешки во инструкции (операции со поинтери)

-Интераптите можат да се предизвикани од корисникот или системот:

-Maskable интерапти – кои може да се оневозможат или

игнорираат

опслужат

-Nonmaskable – интерапти со висок приоритет кои мора да се

1. Што претставува виртуелна машина?

-Виртуелна машина претставува единствен поглед на компјутерскиот систем од гледна точка на корисникот.

1. Како изгледа paper – model на оперативниот ситем на јадрото?



1. Како се врши планирање и распоредување на задачи со помош на ОС

-планирање и распоредување на задачи – одредување на задача спремна за извршување, т.е. кому ќе му се додели главниот процесор.

-јазик за управување на работата – контролни наредби 23.Како се врши елиминирање на зависноста од ВИ операции

-ОС ги изолира бавните ВИ операции од процесорот – со брз медиум за привремено зачувување на сите ВИ уреди

-во минатото лента, а денес диск

-ВИ операции треба да бидат што повеќе поклопени (со други процесорски работи)

-две хиерархиски структури: канал и технички прекин

24.Кои се проблемите на конкурентоста што треба да ја обезбеди ОС?

-Проблеми при преминување од една активност на друга

-Проблеми при заштита на една активност од друга

-Синхронизација на зависни активности 25.Што представува недетерминизам?

-Недетерминирачко однесување (мора да одговори на непредвидливите барања и настани)

-подготвен за сите можни секвенци на случувања 26.Што представуваат hard real системи или soft real-time системи

-Hard real-time системи представуваат хардвер или софтвер кои мора да ги изврши своите операции во одреден временски рок.

-Soft real-time системи се системи чија работа е деградирана ако резултатите не се произведуваат според одреденото назначено време.

1. Кои се концепциите за проектирање на ОС?

-Монолитна организација

-Нивоовска организација

-Архитектура на микројадро

1. Што претставуваат системски повици и како се реализираат?

-Системски повици – аплкативните програми комуницираат со ОС со помош на системски повици, т.е. преку операции кои ги дефинира ОС. Се реализираат со помош на систем на прекини:

-Корисничката програма ги поставува параметрите на системскиот повик на одредени мемориски локации или регистри на процесорот

регистрите

-го иницира прекинот

-оперативниот систем ја презема контролата

-ги зема параметрите

-ги извршува бараните задачи

-резултатот го става на одредени мемориски локации или во

-ја враќа контролата на програмата

1. Што представува јадро на ОС?

-Јадрото е дел на ОС кој што овозможува програмата на сигурен начин да му пристапи на хардверот. Јадрото на оперативниот ситем не е неопходно за стартување и извршување на програмата.

1. Која е основната функција на јадрото?

-Определување на процес т.е. создавање на опкружување во кое може да постојат процеси на процесорите и обезбедување на механизми за интерпроцесна комуникација.

-како на еден процесор во еден момент може да се изврши еден процес

-јадрото одредува кога и на кое време процесот ќе добие процесор.

1. Кои се компонентите кои се неопходни да ги содржи хардверот за да може јадрото да ја оствари својата функција

-Механизам за прекинување

-Заштитен механизам за адресирање на меморија

-Збир на прилевигирани инсртукции

-Часовник за реално време

1. Кои се основните целини за функционирање на јадрото според теоретскиот модел?

-прво ниво за обработка на прекини

-диспечер на системот

-рутини за остварување на интерпроцесни комуникации 33.**Што претставува процес?**

-Процесот е програма или дел од програмата во состојба на извршување заедно со сите ресурси кои се потребни за работата на програмата.

# Кои делови ги содржи?

-програмска или текстуална секција која не се менува и која содржи програмски код

-стек секција која содржи привремени податоци (параметри за процедури, повратни адреси, локални променливи)

-секција на податоци која содржи глобални променливи

# Што опфаќа процесот?

Процесот опфаќа и :

-вредност на програмскиот бројач

-вредност на останатите важни регистри на процесорот, како и

-влезно-излезни ресурси кои што ги користи како и разни други датотеки и влезно-излезни уреди.

# Основниот животен век на процесот = прашање 7 Што претставува контролен блок на процесот?

-За да оперативниот систем знае каде да го продолжи извршувањето, на секој процес се додава пратечка информација т.е. единствен контролен блок. Контролен блок е дел од меморијата односно мемориска структура со основни информации за процесот кој што ОС го користи за управувае со тој процес.

# Што претставува хардверски контролен блок на процесорот?

-Делот на контролниот блок во кој се чува контекстот уште се вика и хардверски контролен блок на процесорот или хардверски дексриптор на процесот.

# Да се опише проширениот дијаграм на состојби на процесите

Процесот може да се најде во неколку состојби (5 или 7 звисно од конечниот автомат кој што е користен за опишување на состојбата), а следните три најважни:

-состојба на извршување (RUN, RUNNING) процесот извршува инструкции на овај процес

-состојба на чекање на процесор (READY, RUNNABLE) процесот ги добил сите потребни ресурси освен процесорот спремен е за работа да му се додели на процесор

-Состојба на чекање на ресурс (WAIT, UNRUNNABLE) процесот чека на некое случување бидејќи за извршувањето на процесот е потребен ресурс **Да се објаснат транзициите**

**-**READY-RUN – Доделувањето на процесорот на процес кој стига на почетокот на редот. Процесот преминува од состојба на чекање на процесор во состојба извршување така што процесорот прекинува извршување на некој процес и започнува извршување на нов

-RUN-READY – Одземање процесор од процесот после истекувањето на периодот во кој што е процесот доделен. Доколку пак оперативниот систем е со претпразнење, процесорот може да се одземе и доколку наиде процес со повисок приоритет. Оваа транзиција е можна само во повеќепроцесни ситеми

-RUN-WAIT – Одземање на процесорот од процесот доколку ресурсот кој е потребен за извршување е зафатен

-SUSPENDED-READY – во оваа состојба процесот доаѓа доколку е суспендиран во состојбата на чекање на процесорот

-SUSPENDED-WAIT – во оваа состојба процесот доаѓа доколку е суспендиран во состојба на чекање на ресурс

# Што претставува работење на системот со претпразнење

**-**Доколку ОС овозможува претпразнење, процесорот може да се одземе доколку наиде на процес со повисок приоритет. Со тоа се обезбедува навремено извршување на критични процеси.

# Како оперативниот систем обезбедува механизми за доделување на процесори на различни процеси?

-Доколку на системот се извршува повеќепроцесен ОС, ОС треба да обезбеди соодветен механизам за доделување на процесори на различни процеси. Делови на тие механизми се :

-редовите на чекање на процесор

-планерите на работи

# Како функционира редот за чекање на процесор?

и обратно

# Што претставува планер на работи?

-Планерот на работите кој во хиерархискиот модел се ноаѓа над јадрото ги извршува следните функции :

-Ги дели работите на процеси

-Врз основа на алгоритми доделува приоритети на процесите

-Ги доведува процесите во ред на чекање за процесор

# Кога се појавува планерот?

-Планерот на работи се повикува само кога ќе се појават нови процеси или кога еден или повеќе завршат

# Што претставува диспечер?

**-**Задача на диспечерскиот систем е да доделува процесор на процесите кои што се наоѓаат во процесоркиот ред.

# Што представува преполнување на системот?

**-**Замената на контекстот претставува чиста загуба на време, односно, преполнување на системот, но заради спроведување на мултипрограмирањето мора да се изврши.

-Преполнувањето на системот зависи од:

-хардверските карактеристики на процесорот

-меморијата

-бројот на регистри на процесорот кои мора да се чуваат

-инструкциите кои се задаваат

-техниките за управување со меморијата

-Ова преполнување е многу помало при користењето на нитки

# Што представува среден распоредувач и функции?

**-**Употребата на swap техниката во интерактивните системи води до воведување на ниво на распоредување на процесите – среден распоредувач.

-Ги извршува следните функции:

-Функцијата на суспендирање на процесот

-Функцијата на враќање на процесот во сотојба на подготвеност

-Изборот на процес за двете функции

1. Кои се релациите помеѓу процес родител и процес дете?

-ги делат сите ресурси

-ги делат подмножеството ресурси на родителскиот процес

-не ги делат ресурсите

1. Како може да биде поделен меморискиот адресен простор?

-Особено може да биди чувствителен меморискиот адресен простор каде се применуваат следните техники:

-процесот дете го дуплира адресниот простор на родителите

-адресниот простор на детето се генерира според програма со која тој адресен простор се полни

1. Како може да заврши процес?

-По правило процесот родител може да го прекине извршувањето на процесот кој што го направил, а тоа се прави од следните причини:

-процесот дете при користењето на некој ресурс ја надминам

квотата

-активноста која процесот дете ја извршува не е веќе потребна

-процесот родител ја завршил активноста пред детето, што се

смета за ненормална ситуација која оперативните системи не ја дозволуваат – и во тој случај сите процеси деца мора присилно да се уништат.

1. Кои се рутините за интерпроцесна комуникација?

-Рутините за интерпроцесна комуникација мора да се имплементирани во јадрото за да бидат достапни за сите процеси и за да имаат директен пристап до диспечерот.

1. Кои се правилата за синхронизација кај процеси?

-Потрошувачот ништо не може да земе од баферот доколку произведувачот претходно не го има ставено во бафер

-Баферот исто така има свои правила. Во идеален случај баферот е со бесконечен капацитет има бесконечен број места и никогаш не може да се наполни. Ваков случај практично не постои.

1. Како функционира деллива меморија?

- од скриптата

1. Како се врши испраќање на пораки?

**-**без употреба на деллива меморија

-се користи дистрибуирана околина (процесите се извршуваат на различни компјутери)

-за да може процесите да комуницираат меѓу себе со испраќање на пораки треба во јадрото да ги применуваат следниве основни операции :

-испраќање на порака

-прием на порака

-пораките кои се разменуваат меѓу процесите можат да бидат со :

-фиксна големина – функционира поедноставно, но не е флексибилна

-променлива големина – обратно

-шеми за реализација на овие операции:

-Синхроно или блокирачко испраќање и примање на пораки

-Асинхроно или неблокирачко испраќање и примање на пораки

-Пораките се примаат преку

-порт

-поштенско сандаче – објект во кој ОС ги остава пораките од други процеси

-секое сандаче си има единствена идентификација

-два процеси за да комуницираат мора да делат сандаче

-меѓу секој пар процеси може да постојат повеќе различни врски каде секоја одговара на едно сандаче.

1. Што претставува семафор и функција?

-Семафор претставува целобројна ненегативна променлива чија вредност штити некој ресурс и овозможува комуникација меѓу процесите. Функции на операциите

-Wait(s) – ги воведуваат процесите во состојба на чекање на

ресурс

ресурс

-Signal(s) – ги изведуваат процесите од состојба на чекање на

-Испраќаат информација до диспечерот за промена во

проецорскиот ред

-Неопходно е да се применат во јадрото на ОС 51.Што претставува socket и негова функција

-Socket(приклучок) представува најчесто користен механизам за размена на пораки преку мрежа за процеси кои се извршуваат на различни компјутери.

1. Што претставуваат лесни процеси? Кои се предностите на програм со повеќе нитки? Која е разликата помеѓу јадрени и кориснички нитки?

-Лесни процеси претставуваат целини за ивршување на кодот кај современите ОС

-Предности од употреба на програми со повеќе нитки:

-Намалување на времето на одзив

-Поефикасно делење на ресурсите и економичност

-Подобро искористување на повеќепроцесорската архитектура

-Разлики помеѓу кориснички и јадрени нитки се:

-За разлика од корисничките нитки, нитките на јадрото не мора да припаѓаат на процес.

-Распоредувањето на процесорот за корисничките нитки го извршува библиотека за работа со корисничките нитки, додека распоредувањето за нитките на јадрото го извршува процесорот.

-Јадрото не е свесно за корисничките нитки – потполна поддршка за работа со кориснички нитки обезбедува посебна библиотека.

1. Кои се концепции за пресликување на корисничките нитки во нитки на јадрото?

-повеќе во една

-една во една

-повеќе во повеќе

1. Кои се критериумите за доделување на процесори?

-Искористување на процесорот

-Пропусна моќ на системот

-Време потребно за комплетирање на процесот

-Време на чекање

-Време на одзив

1. Кои се алгоритмите за доделување на процесори?

-First Come, First Served (FCFS)

-Shortest Job First (SJF)

-Round Robin

-Распоредување врз основа на приоритети

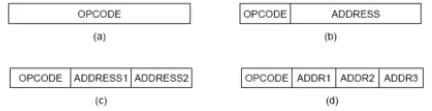
1. Што претставува застој? Дали кај систем што содржи еден процес може да се појави застој?

-Кога процесот бара ресурс, а ресурсот не е расположлив, процесот влегува во состојба на чекање на ресурс (WAIT) и се блокира. Ова појава е состојба на застој или блокада. Кај систем што содржи еден процес не може да се појави застој.

1. Како изгледа форматот на инструкцијата?

-Инструкцијата се состои од код на операција и информации и тоа од каде се земаат операндите и каде се смести резултатот.

1. Што претставува нула-адресен, едно-адресен, дво-адресен и три-адресен инструкциски формат?



1. Како се делат архитектурите според тоа каде се сместени операндите?

-Меморија-меморија – може да имаат 2 или 3 операнди во меморија

-Регистер-меморија – најмалку еден операнд е во регистар, а др. во меморија

-Load-Store – операндите се сместени во регистер

-Стек

60.61.Кои се начините за адресирање?

-Непосредно – кај непосредно адресирање операндот е сместен во самата инструкција

-Директно – операндот е сместен на адреса дадена во инсртукцијата по кодот на операцијата

-Регистерско – содржината на регистерот кој е специфичен со адресата дадена по операцискиот код се користи како операнд

-Индиректно – по кодот на операција е дадена адреса на која се наоѓа физичката адреса на операндот

-Индексно и базно – индекс регистерот се користи за чување на поместување кое се додава на операндот и ја дава ефективната адреса на податокот

-Stack адресирање – со помош на стек можни се нула-адресни инструкции. Се подразбира дека операндот е сместен во стек.

1. Кои се 2-та основни типови на меморија?

-RAM (random access memory) read-write memory

-SRAM – статички РАМ

-синхрона

-асинхрона

-DRAM – динамички РАМ

-Multi bank DRAM

-First Page Mode DRAM

-Extended Data Out(EDO) DRAM

-Burst EDO DRAM

-Synchronous DRAM

-Synchronous-Link DRAM

-Double Data Rate DRAM

-Direct Rambus DRAM

-T-RAM – во развој (комбинација од SRAM и DRAM

-Z-RAM – стопиран

-ROM (read only memory) – чува критична информација за работа на системот, како програм за boot.

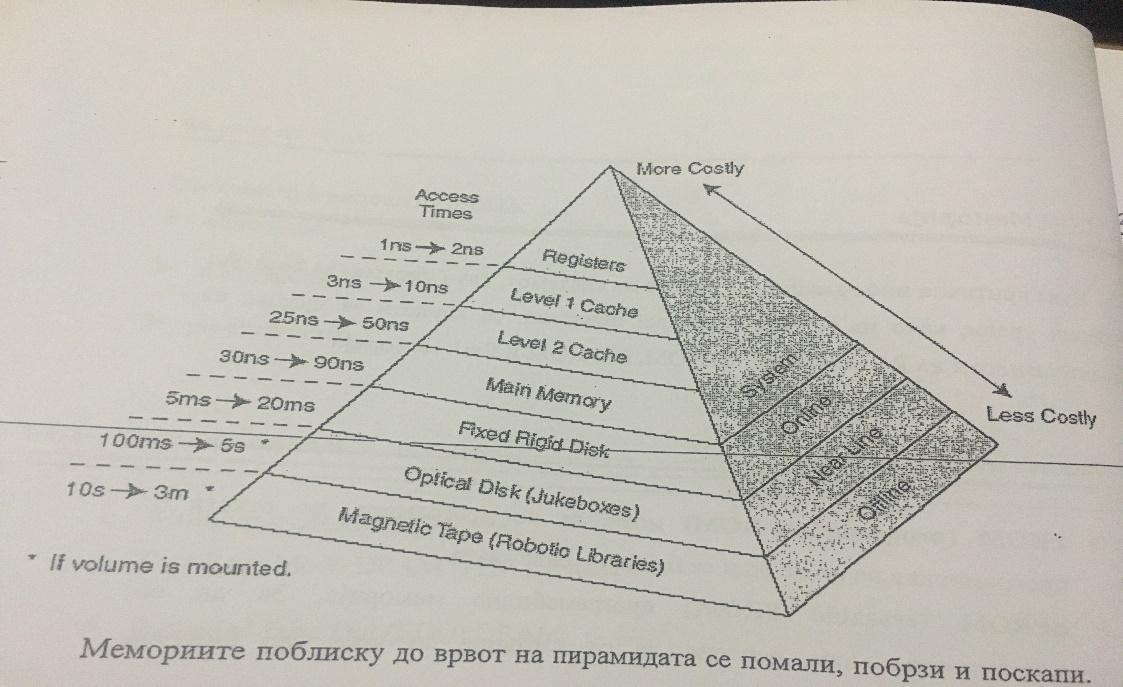
-Programmable ROM (PROM)

-Erasable PROM (EPROM)

-Electrically Erasable PROM (EEPROM)

-Flash

1. Како изгледа хиерархијата на мемории?



1. Врз кои карактеристики функционира хиерархијата на мемории?

-Временска локалност – податоците кои се користени може повторно да станат потребни во блиска иднина

-Просторна локалност – податоците при работа со полиња или јамки формираат кластери во адресниот простор

-Секвенцијална локалност – постои тенденција на инструкциите да им се пристапува секвенционално

1. Кои се термините од хиерархија на мемории?

-Погодок(hit) – бараниот погодок се наоѓа во дадено ниво на мемориска хиерархија

-Промашување(miss) – бараниот податок не се наоѓа на дадено ниво.

-Процент на погодувања(hit rate) – процент на мемориски пристапи за дадено ниво во кои настанува погодок.

- Процент на промашувања(miss rate) – процент на мемориски пристапи за дадено ниво во кои настанува промашување.

-Време на обработка на погодок(hit time) – време потребно за пристап до податок на одредено ниво меморија.

-Време на обработка на промашување(miss penalty) – време потребно за обработка на промашување, време потребно за замена на блок во повисокото ниво на меморија и пренос на податоци до процесорот.

1. Што претставува кеш – меморија?

-Кеш е брза меморија за привремено сместување на податоци од често употребуваните мемориски локации.

1. Што значи byte-адресибилна меморија и збор-адресибилна меморија?

-Ако меморијата е byte-адресибилна секој byte има уникатна адреса, а ако меморијата е збор-адресибилна секој збор има уникатна адреса.

1. Што претставува мемориска единица?

-Мемориска единица е колекција на ќелии кои имаат можност за памтење, заедно со колата кои можат да ја пренесат потребната информација од и кон мемориска единица.

1. Како изгледа блок дијаграмот на една мемориска единица?
2. Кои се целите на управувањето со меморија?

-Алокација на меморијата т.е. доделување на меморијата на процесорите

-Разделување на физичкиот и логичкиот простор на програмата и врзување на адресата

-Логичка организација на меморијата што значи разделување на неизменливите сегменти ( модули и процедури ) од сегментот со непроменлива содржина, односно податоците.

-Релокација, која опфаќа собирање, односно дефрагментирање на работната меморија (врзување на одреден број дисконтинуирани мемориски блокови во еден континуиран нефрагментиран простор) и swap (суспендирање на процесот со негово сместување во дискот)

-Поддршка за динамичко полнење на меморија со програмата и динамичко поврзување.

1. Што значи алокација или доделување на меморија? Што е континуирана алокација? Што е дисконтинуирана алокација?

-Целта на мемориската алокација е што поефикасно доделување на меморијата. Меморијата се дели на најмалку два дела од кои едниот е наменет за резидентниот дел на оперативниот систем, а другиот е наменет за корисничките процеси.

-Континуирана алокација – логичкиот и физичкиот адресен простор на процесорот се состои од низа на мемориски адреси, при што мемориските делови кои се доделуваат на процесите може по големина да бидат еднакви или различни.

-Дисконтинуирана алокација – генерално гледано, физичкиот адресен простор на процесот не е реализиран како континуирана низа на мемориски

адреси. Дисконтинуираната алокација ги опфаќа методите на страничење, сегментација и страничење со сегментација.

1. Како се врши врзување на адреси?

-Поврзување на инструкциите и податоците со мемориски адреси се одвива во следните фази:

-Време на преведување – бидејќи не е познато процесот кој ќе го извршува кодот каде ќе биде сместен во меморијата, преведувачот ги генерира релативните, а не апсолутните адреси. Потоа програмата може да се смести било каде во меморијата.

-Време на вчитување на меморијата – во оваа фаза поврзувачот и полначот на релокатибилниот код ги генерираат апсолутните адреси и ја полнат меморијата со програмата. Поврзувачот може да ја поврзе корисничката програма со други релокатибилни мемории.

-Време на извршување – за време на извршувањето, процесот може да се помести од еден сегмент во друг (вклучувајќи го и дискот, доколку се користи виртуелна меморија)

1. Што претставува единица за управување со меморијата?

-Мапирањето(пресликувањето) на виртуелниот адресен простор во физички го прави хардверскиот уред кој се вика MMU(Memory-Management- Unit) – единица за управување со меморијата.

1. Што претставува релокациски регистер, а што регистер на ограничување?

-Релокациски регистер ја дефинира адресата на физичкиот почеток на програмата и ја содржи најниската адреса на процесорот, а регистерот на ограничување содржи најголем опсег на логички адреси на процесот.

-Релокацискиот регистер и регистерот на ограничување се два регистри на процесорот, кои се полнат тогаш кога процесорот добива процесор за извршување.

1. Кога се врши размена на страници во меморија?

-Процесорот кој се разменува мора да биде потполно ослободен од активностите, не смее да работи, ниту да чека крај на некоја влезно-излезна операција.

1. Кои се програмерските техники за управување со меморијата?

-Динамичко вчитување на програмата

-Динамичко поврзување

-Техника на преклопување

1. Што претставува динамичко вчитување на програмата во меморија и кои се предностите?

-Суштината на динамичко полнење се состои од сместување само на потребните делови на програмата во меморијата, при што се вчитуваат само тогаш кога ќе ги повика програмата.

-Предности

-рутините кои моментално се непотребни не зафаќаат место во

меморијата

-динамичко полнење не бара специјална поддршка од ОС

-ОС може да му помогни на програмерот така што обезбедува

библиотека за динамичко полнење

-лесна измена на системските библиотеки 78.Што претставува статичко поврзување, а што динамичко?

-При статичко поврзување системските библиотеки се третираат како и сите други објектни модули кои се комбинираат со корисничката програма во единствена извршна верзија и со неа се полни меморијата, а кај динамичко поврзување рутината од системските библиотеки се полни по потреба во време на извршување.

1. Која е разликата помеѓу динамичко полнење и динамичко поврзување?

-За разлика од динамичкото полнење, динамичкото поврзување бара подршка на ниво на оперативен систем – истата системска рутина која е во меморијата може да ја користат повеќе процеси.

1. Што претставува техника на преклопување и примери од скриптата?

-За да се овозможи извршување на процесот кој е поголем и од самата физичка меморија се користи техниката на преклопување.

1. Што претставува континуирано доделување на меморија и мултипрограмирање со фиксни партиции?

-Цела меморија се дели на повеќе делови, а процесот се сместува во оној дел кој е еднаков или поголем од меморијата која ја бара процесорот.

-Кога постојат повеќе редови на чекање, поголемиот број на мали процеси може да почека во редот на мали партиции, додека големите партиции не се искористени. Во тој случај има доволно меморија, но не се користи.

1. Што претставува интерна фрагментација?

-Појавата кога деловите на меморијата кои се поголеми од процесот се потполно неискористени

1. Кои се начините на доделување на меморија при динамичко доделување и задача?

-Бит мапи – меморијата се дели на делови со иста големина, а во крајни случаи тие се бајти.

-Поврзани листи – поврзаните листи се градат од записи во следните структури: првото поле означува тип на меморија, второто поче ја содржи почетната адреса на делот на меморијата кој ја опишува дадениот запис. Третото поле ја означува должината на опишаната меморија. Четвртото поле содржи покажувач кон следниот запис.

-Систем на здружени парови – за управување со меморијата се користи по една листа за секој блок слободна меморија со големина 2n B.

1. Кои се алгоритмите за избор на партиција кај партиции со променлива големина? \*

-First Fit

-на процесот му се доделува првата празнина која е доволно

голема

-пребарувањето се одвива или од почетокот на листата на

празнини или од местото до каде сме застанале со претходно пребарување (next fit)

-времето на пребарување е минимално

-се доделува првата празнина на која ќе се наиде, а е доволно голема, без оглед дека понатаму може да има и друга која е поповолна.

\*Едноставен и брз – може да има многу процеси кои се полнат на почетокот на меморијата, бидејќи почнува пребарувањето на најсоодветна слободна празнина.

-Best Fit

-На процесот му се доделува најмалата празнина која е доволно голема за сместување на процесот.

-Се пребарува целата листа

-Ако листата е поделена по големина, не мора да се пребарува

цела празнина

-По сместувањето на процесот, останува најмалата можна

\*Обично е најлош генерално гледано – сместува процес во

најмалиот мемориски блок кој е слободен. Креира минимални фрагменти кои бараа збивање.

-Worst Fit

-На процесот му се доделува најголемата празнина

-Се пребарува целата листа

-По доделувањето, останува најголемата празнина која е доволна за сместување на други големи процеси

\*Сместува во најголемиот неискористен блок

-Next fit – работи полошо од first fit

-Ја пребарува меморијата од локацијата на последниот алоциран блок и го избира следниот слободен блок кој е доволно голем

-Многу почесто се алоцира мемориски блок на крајот на меморијата онаму каде може да се најде најголем блок

-Збивање се извршува со цел да се добие голем блок на крај на

меморијата

1. Што претставува екстерна фрегментација и како се решава? ovde

-Екстерна фрегментација се јавува ако празнините кои настанале се недоволно големи за да се смести некој процес

-Решение

-Суспендирање на активниот процес и негово префрлување на диск. Со тоа се ослободува меморија и се здружува со другите празнини

-Собирање на меморијата 86.Што претставува страничење?

-Страничење е метода со хардверска поддршка на ниво на процесорот која се користи во сите оперативни системи и на сите компјутерски архитектури.

1. Како изгледа логичката адреса кај страничењето?

-Секоја логичка адреса која ја генерира процесорот се дели на два дела:

-број на страници – се користи како индекс во табелата на страници која содржи базна адреса на рамката. Базната адреса го претставува горниот дел на адресата.

-Поместување внатре во страницата - ја дефинира положбата во однос на самата страница и во комбинација со базната адреса – дефинира полна физичка адреса која се испраќа во мемориската единица. Основната адресибилна единица е бајт и поместувањето е исто и за логичката и за физичката адреса/

1. Како хардверот го поддржува страничењето?

-Со посебен пар на регистри

-PTBR(page-table base register) – покажувач на табели на

страници

-PRLR(page-table length register) – покажувач на големината на

табелите на страниците.

-TLB – многу брза компонента, релизирана на принцип на асоцијативна меморија. Секој влез се состои од клуч и вредност.

1. Што претставува сегментација?

-Сегментација е метод за управување со меморијата која поддржува логички кориснички поглед врз меморијата.

1. Како изгледа логичката адреса кај сегментацијата

-Се состои од два дела:

-име на сегментот – обично се задава број кој претставуа идентификатор на сегментот.

-поместување внатре во сегментот 91.Што претставува сегментација со страничење ?

-Страничењето ја поништува екстерната фрегментација

-Логичката адреса се состои од:

-идентификатор на сегментот(selector)

-поместување во рамката на сегментот(offset) 92.Како изгледа виртуелната адреса?

-број на сегментот: индексирање на табела на сегменти

-број на страница: индексирање на табелата на страници

-поместување(офсет): да се лоцира зборот внатре во рамката 93.Што претставува виртуелна меморија?

-Виртуелната меморија како и кеширањето е концепт карактеристичен за хиерархијата на мемории. Се користи дискот како проширување на РАМ меморијата и со тоа се зголемува располижливиот адресен простор кој процесот може да го користи. Областа на дискот каде се сместуваат блокови се нарекува page file. Се имплементира со страничење

1. Кои се поимите од виртуелна меморија?

-Виртуелна адреса – логичка или програмска адреса која ја користи процесот. CPU генерира адреса од виртуелниот адресен простор.

-Физичка адреса – реална адреса од физичка меморија

-Мапирање – механизам за преведување на виртуелните адреси во физички

-Рамки на страници – блокови со иста големина на кои е поделена главната меморија

-Страници – блокови на кои е поделена виртуелната меморија со иста големина со страничните рамки. Виртуелните страници се сместени на диск се додека не станат потребни за обработка.

-Страничење – процес на копирање на виртуелна страница од диск во сртанична рамка од меморија

-Странична грешка – настан кој се јавува кога бараната страница не е во меморија и мора да се копира од диск во меморија.

1. Како се одвива страничење кај виртуелна меморија?

-Секој процес има табела на страници( page table ) која се наоѓа во главната меморија и ги чува физичките локации на виртуелните страници на процесорот.

-Табелата на страници има редови колку што има виртуелни страници за процесот. Ако страниците на процесот се во меморија битот за валидност на тие страници е поставен на 1.

-Табелата на страници има 2 полиња: бит за валидност и број на рамка 96.Кои се проблемите при избирање на големина на страница?

-Ако страницата е поголема тогаш табелата на страници е помала и се заштедува простор во меморија

-Ако страницата е голема интерната фрагментација доведува до поголеми загуби на простор

-Поголеми страници овозможуваат помалку трансфери од диск во главна меморија.

-Ако страниците се големи се губи принципот на локалност, бидејќи се пренесуваат податоци кои може да се непотребни.

1. Предности и недостатоци на страничење и виртуелната меморија

-Предности:

-Програмите не се ограничени со големината на физичката меморија. Може повеќе програми да се изведуваат паралелно со што се добива поголема искористеност на CPU.

-Страничењето овозможува оперативниот систем да специфицира заштита и делење на податоци од ниво на страна.

-Недостатоци:

-Виртуелната меморија имплементирана преку страничење додава дополнително мемориско обраќање

-Со користење на TLB времето за пристап се намалува но сеуште се троши време за преведување на број на страница во број на физичка рамка.

-Друг недостаток на виртуелната меморија е трошење на простор во главната меморија за сместување на табела на страни.

1. Кои се поимите од кеширање?

-Кеш погодок (hit) ако бараните податоци се наоѓаат во кеш меморијата

-Кеш промашување (miss) ако податоците не се во кеш меморијата

-ладно промашување(cold miss) -кешот е празен

-конфликтно промашување(conflict miss) – кеш линијата е зафатена од различна мемориска локација

-промашување на капацитетот – работното множество е поголемо од капацитетот

-Политика на поставување – онаму каде е поставен нов блок

-Политика на размена – контролира кој блок е селектиран за исфрлање 99.Кои се алгоритмите за замена кај виртуелната меморија?

-Оптимална политика – за замена се избираат страници за кои времето за следното обраќање е најдолго

-Least recently used (LRU) – замена на онаа страница која е најмалку користена во последниот период

-FIFO (First In First Out)

-Clock – алгоритам ,,втора шанса” – варијанта на FIFO која ја зема во предвид искористеноста на страницата.

1. Кои се функциите на влезно-излезниот подсистем?

-Ги управува влезно-излезните уреди и операции кои тие уреди ги извршуваат и ги контролираат

-Обезбедува што поедноставен интерфејс кон корисникот и остатокот од системот

1. Што претставува независност на уредите?

-Независност на програмата од конкретниот модел на уредот кој се користи – различните модели на уредот од ист тип од аспект на програмата мораат еднакво да се набљудуваат

-Независност на програмата од конкретниот вид на уред кој се користи – на пр. Мора да се обезбеди програмот да можи да ги прочита податоците од дискот или ЦД уредот на што е можно поидентичен начин.

1. Која е последицата од независност на уредите?

-Програмите не работат со реални, конкретни уреди, туку сите влезно- излезни операции ги обавуваат со помош на виртуелни уреди.

1. Што претставува независност на знаковниот код?

-Очигледно, непожелно е да се пишуваат програми кои бараат од корисникот детално познавање на знаковниот код, користен во конкретните уреди. Влезно-излезниот подсистем мора да преземе грижа за препознавање на конкретниот код и за претставување на податоците на корисникот во стандарден облик.

1. Како се класифицирани уредите?

-Според намена на уредите

-уреди за долготрајно сместување на податоците

-уреди за пренос на податоци

-уреди кои обезбедуваат интерфејс кон корисникот

-Според насока на преносот

-влезни

-излезни

-влезно-излезни

1. Како работат контролерите кај влезно-излезните уреди?

-На три начини:

-како приклучок (Port) – пр.Сериски контролер

-како магистрала (Bus) - SCSI

-како уред (device) – пр. Графички уред

1. Како може да се пристапи до уредите во зависност од начинот на организација на меморискиот и влезно-излезниот простор?

-Доколку меморискиот и влезно-излезниот простор се разделени, процесорот има посебни инструкции за запис и читање на податоците од одредена адреса на контролерот.

-Доколку влезно-излезниот простор може да се третира како меморија, трансферите можат да се извршат со обични инструкции за работата со меморија.

1. Како изгледа типичен влезно-излезен приклучок?

-Контролен регистер – служи за поставување режим на работата на уредот како што се брзината на приклучокот и типот на комуникација. Во овој регистер, процесорот исклучиво запишува податоци.

-Статусен регистер – овој регистер го опишува статусот на командите што ги извршува, односно дали командата е извршена, дали е спремен податокот, дали настанала грешка итн. Процесорот од статусниот регистар исклучиво ги чита податоците

-Регистер на податоци за влезен режим – влезниот регистар служи за читање на податоци од влезните уреди. Процесорот од влезниот регистар исклучиво и ги чита податоците.

-Регистер податоци за излезен режим – излезниот регистар служи за запишување на податоците во излезниот уред. Од овај регистар процесорот исклучиво информации.

1. Што претставува техника на повлекување на уредите?

-Повлекување на уредите се извршува во јамка:

-компјутерот го повторува читањето на статусниот регистар

-го анализира busy битот, се додека вредноста на битот не стане 0

-Основен недостаток е трошењето на процесорското време на јазолот за прозивање.

1. Што претставува механизмот за прекин?

-Недостатокот може да се отстрани со воведување на хардверски механизам кој овозможува уредот да му означи на процесорот кога е командата завршена. Овој механизам се нарекува прекин.

1. Кои се карактеристиките на мемеханизмот за прекини?

-можност за одложување на обработката на прекинот додека процесот е во критична состојба

-брза и ефикасна техника на одредување на уредот кој што го поставил прекинот

-овозможено е вгнездување на прекинот

-повеќе-слоен прекинувачки систем, кои ќе го разликуваат приоритетот на прекинувачките сигнали и по приоритет ќе ги решаваат прекините.

1. Како се вика хардверскиот уред кој ги реализира прекините?

-Приоритетен прекинувачки контролер (PIC) 110.Што значи маскирна, а што немаскирна линија ?

-Немаскирна линија

-Прекинувачкиот сигнал секогаш може да го прекини извршувањето на тековниот процес заради критичните хардверски грешки како што се грешките во меморијата

-Маскирна линија

-не го прекинува извршувањето на процесот, додека процесот е во критична секција.

-за праќање на сигнали нормалните операции и стандардните грешки кои се јавуваат на ВИ уреди

1. Што претставува DMA контролерот за директен пристап до меморија?

-DMA се користи за да се избегне програмирањето на ВИ за трансфер на големи количини на податоци

-DMA контролер – за да може да се направи брз пренос на податоци кај уредите како што е дискот, на релација контролер-работна меморија и обратно.

1. Како се обезбедува униформен интерфејс кон апликациите

-Сродните уреди се групирани во класи, а драјверите ги апстрахираат карактеристиките на конкретните уреди. На програмерот понатаму му се нуди стандарден склоп на системски повици за работа со влезно-излезните уреди.

1. Што претставува кеширање?

-Кеш(cache) претставува област на брза системска меморија која ја чува копијата на податоците, најчесто од дискот. Пристапот на податоците во кешот е значително побрз од пристапот кон податоците на ВИ уреди.

1. Што претставува бафер, а што спулер?

-Бафер е дел од меморијата која функционира на принцип производител-потрошувач и служи за чување на привремени податоци при пренос на податоците помеѓу два уреди, или помеѓу уред и апликација, а спулер е бафер кој привремено ги чува излезните податоци наменети за некој неделлив уред

1. Која е разликаат помеѓу кеш и бафер?

-баферот моментално ги чува актуелните податоци, а кешот чува било која копија од дискот

1. Кои се целите на баферирањето?

-Израмнување на различните брзи помеѓу потрошувачот и производителот

-Прилагодување на различни количини на трансфер на податоци

-Оддржување на семантика на копирање

1. Што претставува заштита и сигурност на оперативниот систем?

-Проблемот на заштита на оперативниот систем се однесува на контрола на пристапите и внатрешноста на системите и се сведува на

контрола на пристапот на програмите, процесите и корисниците на ресурсите на оперативниот систем

-Сигурноста опфаќа повеќе аспекти и вклучува интеракција помеѓу системите и надворешниот свет. Нарушувањето на сигурноста може да биди случајно и намерно.

1. Што значи сигурност?

-Сигурноста опфаќа заштита на системите од :

-Неовластен пристап кон податоци и ресурси

-Злонамерна измена на податоци

-Злонамерно уништување на податоци

-Спречување на легитимно користење на системот 119.Што претставува домен на заштита?

-Доменот на заштита претставуа колекција од права на пристап на дефинирани парови(име на објект, множество на права)

1. Како се реализира матрицата на право на пристап?

-Заштита може да се прикаже како матрица на пристап во која:

-редовите претставуваат домени, а

-колоните претставуваат објекти

-Елементот на матрицата(i,j) претставува збир на операции кои процесот од доменот Di може да ги изврши над објектот Oj.

1. Кои операции можат да се спроведат во матрицата на право на пристап?

-Операција copy – со операцијата copy се копира право над објектот, при што одредишното поле припаѓа на иста колона.

-Право на сопственост(owner) – во матрицата е потребно да се воведе механизам кој овозможува додавање на нови права или укинување на постоечките.

-Право на контрола на доменот – операциите копирање, доделување и одземање на права ја модификуваат содржината на одредени колони во матрицата.

1. Како се спроведува механизмот на клучеви?

-компромис помеѓу глобална табела и листа за контрола на пристапот на објектите

-на секој објект му се доделува листа на брави(lock), а на секој домен листа на клучеви(keys).

-клучевите и бравите се единствени низи на битови. Процесот од доменот може да пристапи на објектите само ако неговиот клуч не одговара на една од бравите на објектот.

-овој механизам е флексибилен и ефикасен, зависно од големината на клучевите. Правата може да се одземат ако се изменат битовите кои ја креираат бравата.

1. Кои се аспектите на безбедност?

-Физичко ниво

-Човечки фактор

-Мрежно ниво

-Ниво на оперативен систем

1. Како се врши идентификација на корисниците?

-специјален хардвер како што е клуч или ID картичка

-внесување на доверливи информации, како што е лозинката

-биолошките атрибути на корисникот(отпечаток од прст, потпис, снимање на очната рожница)

1. Методи на напад

-Одбивање на услуга – стопирање на работата на програмата или сервисот така што другите корисници не можат да работат со нив

-Лажирање(spoofing) на DNS – напаѓачот ги прати IP адресите во IP пакетите и се претставува како друг компјутер.

-Smurf – напаѓачот преку интранет им праќа на сите членови на мрежата ICMP ECHO-REQUEST пакети и како одредишна IP адреса ја става адресата на жртвата.

-Нушкање (sniffing) - метод во кој со специјална програма наречена нушкач(sniffer) се пресретнуваат TCP/IP пакети кои поминуваат низ одреден компјутер и по потреба се прегледува нивната содржина.

-Смртоносен ping (Ping-of-Death) – праќање на голем број на ICMP REQUEST пакети може да биде фатално за некои оперативни системи.

-Скенирање на приклучоци – служи за идентификација на отворените приклучоци на жртвата – напаѓачот праќа SYN или FIN пакети кон опсегот на приклучоци и очекува RST пакети за неактивните приклучоци.

1. Кои се програмски закани?

-Програмски закани – програмата која ја напишал еден корисник (програмер), а со која се служат други корисници претставува потенцијална закана.

-Тројански коњи - тројанските коњи се нелегален сегмент на код подметнат во кодот на програмата со цел да се промени функцијата или однесувањето на оригиналната програм

-Замка (trap door) – авторот на програмата може сакајќи или ненамерно да остави празно место во својот код. Оној кој ги знае тие места може да подметне свој код и со тоа да оствари добивка

-Преполнување на стекот или баферот – најчесто упатуван напад од мрежа при обид за неовластен пристап кон системот.Овој тип на напад може да го користат и овластените корисници со цел да го измамат системот и да остварат поголеми права од оние кои ги имаат.

1. Кои се системски закани?

-Вирус – претставува вид на системска закана која може да изврши различни деструктивни активности како што е бришење датотеки, цели дискови и слично

-Компјутерски црв – процес кој користи механизам за размножување со цел да ги деградира системските перформанси

\*Морисов црв – се состои од главна програма која се обидува да ги открие лозинките на корисникот и јадица(grappling hook).

1. Кои се техниките за зголемување на сигурноста на системот?

-Потребно е да се провери дали постојат:

-кратки лозинки или лозинки кои лесно се погодуваат

-опасни програми со доменски бит

-неавторизирани програми во системскиот директориум

-неочекувани процеси кои долго време се извршуваат

-несоодветна заштита на директориумот

-несоодветна заштита за системските директориуми

-опасни влезови во програмската патека

-промени во контролните збирови на системските програми 129.Како работат мрежните бариери?

-Мрежната бариера го филтрира сообраќајот на релација локална мрежа-Интернет. Мрежните бариери користат три основни методи за заштита на мрежно, транспортно и апликациско ниво.

130.Што значи откривање на напади?

-IDS(Instruction Detection System) цели кон откривање на напади и реакции на обид за напад на системот.

-Методите на детекција може да се класифицираат во две групи:

-Детекција која се базира на ознаки – каде IDS анализира мрежен сообраќај и бара карактеристичен пример кој го открива нападот, како што е типична секвенца на повеќеструко прогрешно пријавување

-Детекција на аномалии – каде IDS открива аномалии како што се системски повици со големо количество влезни податоци кои укажуваат на обид за напад преку преполнување на меѓумеморијата-buffer.