Slovenská technická univerzita v Bratislave
FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLÓGIÍ
Študijný program: POČÍTAČOVÉ SYSTÉMY A SIETE

Bc. Matúš Kováčik

Štúdia pokročilých vlastností operačného systému Solaris

Diplomová práca

Vedúci diplomového projektu: **Ing. Juraj Štefanovič, PhD.**
december, 2007

Prehlásenie

Prehlasujem, že som túto prácu vytvoril samostatne bez ďalšej pomoci, ktorá nieje vyznačená v dokumentácii.

Všetky informácie prevzaté z iných zdrojov sú v dokumentácii označené.

…….....

Matúš Kováčik

ANOTÁCIA

Slovenská technická univerzita v Bratislave
FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLÓGIÍ
Študijný program: POČÍTAČOVÉ SYSTÉMY A SIETE
Autor: Bc. Matúš Kováčik
Diplomová práca: Štúdia pokročilých vlastností operačného systému Solaris
Vedenie diplomovej práce: Ing. Juraj Štefanovič, PhD.
december, 2007

Projekt sa zaoberá analýzou operačného systému Solaris 10 a jeho vybraných pokročilých vlastností. Zameriava sa najmä na oblasti:

- Solaris Zones
- Solaris Containers
- Zettabyte File System (ZFS)
- Bezpečnostné prvky

Skúma ich najmä z hľadiska administrácie, bezpečnosti, stability a konfigurovateľnosti. Na príklade prípadovej štúdie ukazuje praktické skúsenosti s týmto systémom. Pri výskume sa vychádzalo z oficiálnej dokumentácie dodávanej výrobcom, praktických skúseností, testov a experimentov autora, ako aj skúmaných ohlasov komunity profesionálov používajúcich operačný systém Solaris.

Okrem samotného systému je analyzované prostredie operačných systémov, požiadavky na ne kladené. Navrhuje a diskutuje rôzne metódy hodnotenia a porovnávania operačných systémov a uvádza ich príklady pri hodnotení operačného systému Solaris.

ANNOTATION

Slovak University of Technology Bratislava
FACULTY OF INFORMATICS AND INFORMATION TECHNOLOGIES
Degree Course: COMPUTER SYSTEMS AND NETWORKS
Author: Bc. Matúš Kováčik
Thesis: Study of operating system Solaris advanced features
Supervisor: Ing. Juraj Štefanovič, PhD.
2007, December

Projct concerns about analysys of operatinf system Solaris and its chosen advanced features. Project is mainly focused to:

- Solaris Zones
- Solaris Containers
- Zettabyte File System (ZFS)
- Security aspects

This features are explored mainly from point of view of administration, security, stability and configurability. On example of case study illustrates practical experience with this system. Study was based on producer's official documentation, author's practical experience and tests as much as opinions of Solaris professionals community.

Except exploring the system also environment of operating systems and requirements for operating systems are analyzed. Various methods of operating sytems evaluation and comparsion are proposed. Examples of this methods are used to evaluate Solaris operating system.

Obsah

Obsah....5
1. Úvod....8
1.1. Použité skratky....9
1.2. Slovník pojmov....9
1.3. Použitá notácia....10
2. Analýza....11
2.1. Serverové operačné systémy....11
2.2. Operačný systém Solaris....12
2.2.1. Kľúčové vlastnosti....12
2.2.2. Projekt OpenSolaris....15
2.2.3. Výhody a nevýhody OS Solaris....15
2.3. Porovnanie vlastností jadier operačných systémov Linux, Solaris a FreeBSD....16
2.4. Solaris kontajnery....22
2.4.1. Správa úloh....22
2.4.2. Projekty....24
2.4.3. Fair Share Scheduler....26
2.4.4. Ovládače zdrojov....29
2.5. Solaris zóny....32
2.5.1. Základné informácie o zónach....33
2.5.2. Rozdiel medzi kontajnermi a zónami....35
2.5.3. Správa zdrojov....35
2.6. Balíčkovací systém OS Solaris....36
2.6.1. Princíp....37
2.6.2. Zdroje balíčkov....39
2.6.3. Aktualizácie....40
2.6.4. Balíčkovací systém projektu Indiana....40
2.7. Zettabyte File System....41
2.7.1. Storage pools....42
2.7.2. Teoretické limity....44
2.7.3. Transakčný model Copy-on-write....44
2.7.4. Snímkovanie (snapshots) a klonovanie (clones)....45
2.7.5. Dynamické rozloženie záťaže....45
2.7.6. Rýchle vytváranie súborových systémov (partícií)....46
2.7.7. Ďalšie vlastnosti....46
2.8. Bezpečnostné prvky OS Solaris....47
2.8.1. Izolácia aplikácií pomocou Solaris zón....47

2.8.2. Solaris Cryptographic Framework....47
2.8.3. Integrovaný firewall....48
2.8.4. Prednastavená bezpečnosť (security by default)....48
2.8.5. Bezpečnostné rozšírenie Solaris Trusted Extension....49
2.8.6. Automatizovaná správa bezpečnostných záplat....50
2.8.7. Prevencia útokov so zneužitím chyby pretečenia zásobníka....51
2.8.8. Integrita súborov a bezpečné spúšťanie procesov....51
3. Posudzovanie operačných systémov....52
3.1. Riadiaci versus technickí pracovníci....52
3.2. Metódy porovnávania aspektov kvality operačných systémov....52
3.2.1. Porovnávacie merania....53
3.2.2. Porovnávanie zverejnených informácii....55
3.2.3. Vlastné testovanie....56
3.2.4. Názory používateľov....56
3.3. Aspekty kvality operačných systémov....57
3.4. Diskutabilnosť výsledkov merania....61
4. Návrh....63
4.1. Výstupy projektu....63
4.2. Použité metódy....63
4.3. Postup prác na projekte....64
4.4. Nástroj na porovnávacie meranie výkonu operačného systému....65
4.4.1. Požiadavky....65
5. Implementácia....66
5.1. benchmark - nástroj na porovnávacie meranie výkonu operačného systému....66
5.1.1. Princíp činnosti....66
5.1.2. Kompatibilita....68
5.1.3. Popis zdrojových kódov....68
5.1.4. Možnosti rozšírenia....69
5.2. Používateľské návody....69
5.3. Post-inštalačný skript....70
5.3.1. Skript ošetrujúci závislosti originálnych balíčkov....70
6. Hodnotenia operačného systému Solaris....72
6.1. Porovnanie bezpečnosti....72
6.1.1. Operating system vulnerability scorecard....72
6.1.2. Symantec Internet Security Threat Report....74
6.1.3. Secunia.com....75
6.1.4. IBM Internet Security Systems X-Force Trend Statistics....75
6.2. Porovnanie rýchlosti....76
6.2.1. MySQL Innodb SysBench OLTP Benchmark Test....76
6.3. Vlastné meranie rýchlosti....76

6.3.1. Porovnávané operačné systémy....76
6.3.2. Testovacie prostredie....77
6.3.3. Výsledky testov....78
6.4. Názory používateľov na operačný systém Solaris....80
6.5. Prieskum názorov systémových administrátorov....81
6.5.1. Priebeh prieskumu....81
6.5.2. Otázky a vyhodnotenie....82
6.5.3. Náklady na prácu systémového adminsitrátora....91
6.6. Vlastné skúsenosti....92
6.6.1. Realizované operácie....93
6.6.2. Pozitívne hodnotené aspekty OS Solaris....95
6.6.3. Negatívne hodnotené aspekty OS Solaris....95
6.6.4. Komentár....96
7. Záver....97
7.1. Práce zrealizované v rámci projektu....97
7.2. Zhodnotenie....98
Použitá literatúra....100
Zoznam príloh:....102
Príloha č. 1. - Návod na konfiguráciu a správu zón....103
Príloha č. 2 - Návod na konfiguráciu a správu súborového systému ZFS. .104
Príloha č. 3 - Návod na aktiváciu a nastavenie sieťového filtra (firewall) a prekladu sieťových adries (NAT)....105
Príloha č. 4 - Návod na inštaláciu a konfiguráciu systému opensource balíčkov....106
Príloha č. 5 - Návod na aktualizáciu a distribúciu záplat (patch)....107
Príloha č. 6 - Tipy a triky....108
Príloha č. 7 - Návod na post-inštalačnú a konfiguračnú procedúru....109
Príloha č. 8 - Návod na použitie skriptu sundepend....110
Príloha č. 9 - Návod na použitie programu OS benchmark....111
Elektronické médium....112

1. Úvod

Spoločnosť Sun Microsystems vyhlasuje, že operačný systém Solaris 10 je najviac pokročilý operačný systém na svete. Tento projekt si kladie za cieľ preskúmať vybrané vlastnosti tohto systému, ako aj systém ako celok. Systém je skúmaný najmä z hľadiska:

- administrácie
- bezpečnosti
- stability
- konfigurovateľnosti

S verziou 10 priniesol tento operačný systém niekoľko zásadných zmien, ktoré rozprúdili diskusie a vzbudili veľký záujem v komunite systémových administrátorov. Skúmanie sa preto venuje najmä tejto verzii. Špeciálny dôraz je kladený na jeho pokročilé a z technického hľadiska zaujímavé vlastnosti:

- Solaris zóny (Solaris Zones)
- Solaris kontajnery (Solaris Containers)
- Súborový systém ZFS (Zettabyte File System)

Pre potreby skúmania kvality operačného systému bude potrebné preskúmať aj prostredie a požiadavky v súčasnosti kladené na operačné systémy.

Tento projekt si kladie za cieľ vygenerovať a zozbierať relevantné technické informácie zaujímavé najmä pre technických odborníkov, ktorí ale ešte neprišli do styku s operačným systémom Solaris, prípadne sa zaoberajú pokročilými aspektmi správy rozľahlých a zložitých informačných systémov.

Nosná časť tejto práce sa pohybuje v teoretickej rovine. Na rozdiel od väčšiny diplomových projektov, ktorých výstupmi sú softvérové či hardvérové produkty, ktoré ale z pochopiteľných príčin väčšinou nenachádzajú praktické uplatnenie, táto práca má za cieľ vytvoriť publikovateľný študijný a metodický materiál pokrývajúci niekoľko súvisiacich oblastí a prínosný pre záujemcov o ne. Výstupmi projektu by mali byť:

- materiál opisujúci operačný systém Solaris, jeho výhody, nevýhody a princípy činnosti jeho vybraných pokročilých vlastností,

- všeobecná metodika hodnotenia a výberu operačného systému,
- kvantitatívne porovnanie operačného systému Solaris s konkurenčnými produktmi,
- analýza prostredia, pre ktoré je dôležitý výber operačného systému,
- príručka jednoducho popisujúca vybrané konfigurácie operačného systému Solaris.

V rámci práce na projekte by mal byť skúmaný operačný systém nainštalovaný, sfunkčnený, podrobne preskúmaný a dlhodobo testovaný v reálnej prevádzke s využitím čo najväčšieho množstva ním poskytovaných funkcií.

1.1. Použité skratky

OS – Operačný systém

ZFS – súborový systém Zettabyte File System

BSD – Berkley Software Distribution – typ UNIXového operačného systému

API - Application Programming Interface - rozhranie pre programovanie aplikácií

1.2. Slovník pojmov

Operačný systém - je základné programové vybavenie počítača. Je to rozhranie, ktorým používateľ prostredníctvom aplikácií komunikuje s hardvérom.

Cluster – skupina pevne spriahnutýchch spoločne pracujúcich počítačov. Niekedy sa navonok javia ako jediný počítač.

Virtualizácia - proces alebo technika, pri ktorej je nahradený fyzický prostriedok softvérovou vrstvou, ktorá ho emuluje. Takýto prostriedok je pre systém transparentne definovaný, i keď fyzicky neexistuje.

Jadro operačného systému – základný komponent operačného systému zabezpečujúci elementárne operácie s hardvérom, správu pamäte, procesov, súborového systému a mnohé ďalšie elementárne či pokročilé operácie.

Open source – open-source software (OSS) je počítačový softvér s otvoreným zdrojovým kódom. Otvorenosť tu znamená ako technickú dostupnosť kódu, tak aj

legálnu dostupnosť - licenciu softvéru, ktorá umožňuje, pri dodržaní istých podmienok, užívateľom zdrojový kód využívať, napríklad prehliadať a upravovať. opensource softvér v súvislosti s OS Solaris znamená Open source softvér, ktorý bol vytvorený, alebo prispôsobený komunitou tvorcov Open source softvéru pre operačný systém Solaris.

Portovanie – prispôsobenie; v kontexte operačných systémov najmä prispôsobenie aplikácie na beh pod daným operačným systémom.

Skript – je počítačový program určený na uľahčenie (zautomatizovanie) operácií v počítači. Skript je väčšinou počas behu interpretovaný interpretrom priamo zo svojho zdrojového kódu (nekompiluje sa).

1.3. Použitá notácia

Kapitola pojednáva o - normálny text

Predictive Self Healing - odborný názov, názov spoločnosti alebo názov technického komponentu

`task_struct` - zdrojový kód, príkaz, premenná alebo iný systémový názov

je ponúkaný bezplatne – dôležitá informácia

Solaris kontajnery (Solaris containers) – text v zátvorke reprezentuje najmä anglický všeobecne zaužívaný ekvivalent pre text pred zátvorkou, prípadne upresnenie textu pred zátvorkou

2. Analýza

Kapitola pojednáva o operačnom systéme Solaris 10 ako celku a analyzuje ho z rôznych pohľadov. Nachádza sa v nej porovnanie vlastností jadier troch najznámejších UNIXových operačných systémov. Na jej začiatku sú vymenované a stručne popísané zaujímavé vlastnosti OS Solaris, pričom niektoré z nich sú v ďalších častiach podrobne analyzované.

2.1. Serverové operačné systémy

Serverové operačné systémy sú od základu navrhnuté tak, aby poskytovali platformu pre viacužívateľské a mnohokrát kritické sieťové aplikácie. Ako také, sú zamerané skôr na bezpečnosť, stabilitu a kolaboratívne vlastnosti, než na priateľskosť používateľského rozhrania[13].

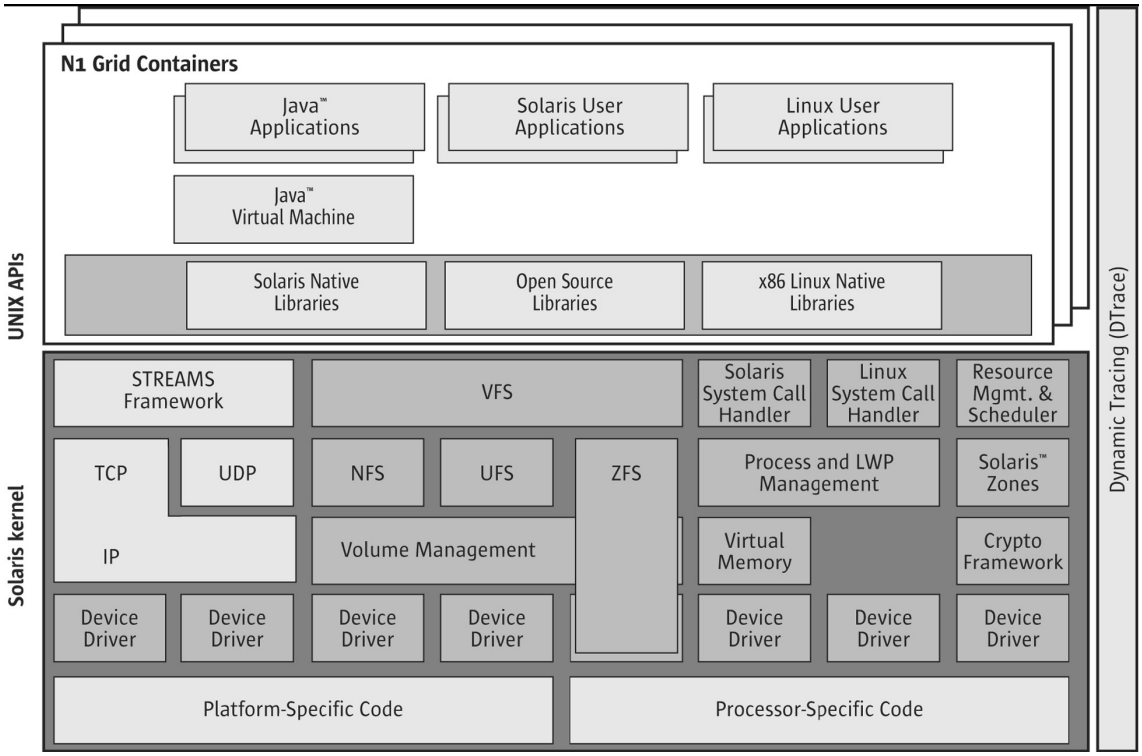
Serverové operačné systémy poskytujú platformu pre viacužívateľské aplikácie a väčšinou sú dodávané spolu s bežnými serverovými aplikáciami, ako napríklad webový server, rozšírené aplikačné servery, servery a klienti elektronickej pošty, služby pre vzdialený prístup a manažment.

Podľa časopisu PC World [13] sú najdôležitejšími vlastnosťami serverových operačných systémov:

- jednoduchosť správy
- bezpečnosť
- stabilita
- vlastnosti (features)
- výkon
- hardvérové požiadavky, hardvérová kompatibilita
- rozšíriteľnosť
- náklady

2.2. Operačný systém Solaris

Výrazné rozširovanie systému Solaris 10 vyústilo aj do čoraz väčšej dostupnosti aplikácií pre tento operačný systém. Systém Solaris 10 dosiahol 125 výkonnostných svetových rekordov a je podporovaný na 785 systémoch, z ktorých 710 je postavených na platforme x64/x86 a ktoré sú vyrábané spoločnosťami Sun, Dell, HP a IBM[1]. Jednou z hlavných strategických výhod systému Solaris 10 je, že j**e ponúkaný bezplatne** a má otvorený zdrojový kód, ktorý je dostupný cez Open Solaris komunitu. Komunita má v súčasnosti[2] 16 400 registrovaných členov, ktorý sú zaradený v 38 používateľských skupinách.



Obrázok 1: Vnútorná štruktúra operačného systému Solaris 10.

2.2.1. Kľúčové vlastnosti

Medzi kľúčové vlastnosti operačného systému Solaris patria[3]:

- Dtrace
- Solaris Containers

1 Údaje k novembru 2006

2 November 2006

3 Uvedené názvy vychýdzajú z pomenovaní stanovených výrobcom

- Process Rights Management
- Predictive Self Healing
- Linux Application Environment
- Zettabyte File System (ZFS)
- Cryptographic Framework
- Sun Update Connection
- AMD64
- Solaris Application Guarantee

Dtrace

S novými diagnostickými nástrojmi v systéme 10 môžu vývojári rýchlo identifikovať výkonnostné problémy a ťažko odhaliteľné chyby, takže problémy možno diagnostikovať za milisekundy a minúty a nie hodiny alebo dni.

Solaris Containers

Solaris 10 je jediným operačným systémom, ktorý poskytuje viacero softvérových partícií s viac ako 8000 kontajnermi v jedinej inštancii operačného systému. Zdroje možno automaticky realokovať, pričom sa dosahuje až 80-percentné využitie systému.

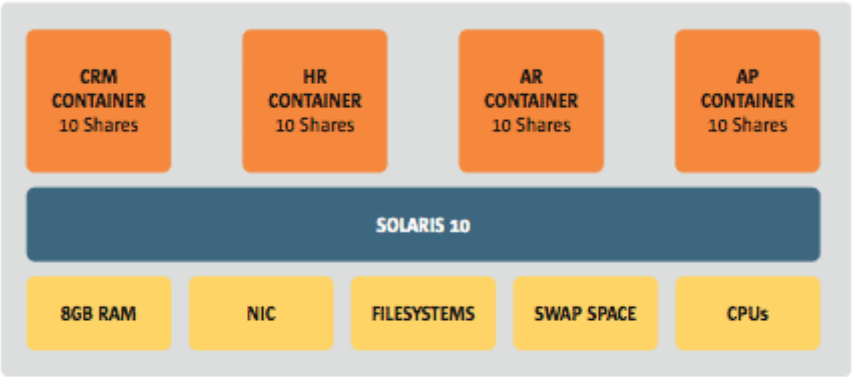


Figure 3—Example configuration

Obrázok 2: Príklad systému so štyrmi zónami

Process Rights Management

Solaris 10 obsahuje technológiu z operačného systému Trusted Solaris, ktorá má rozsiahle využitie v bezpečnej komunikácii štátnej správy USA. *Process Rights Management* využíva jemne členený model používateľských privilégií, čím pomáha

zabezpečiť ochranu pred zneužitím zraniteľných miest aplikácií, ktoré by mohlo viesť k rozsiahlemu poškodeniu systému.

Predictive Self Healing

Solaris 10 dokáže automaticky diagnostikovať, izolovať a zotaviť sa z mnohých hardvérových a aplikačných porúch, čím výrazne znižuje čas systémových výpadkov.

Linux Application Environment

Predtým vystupoval pod kódovým názvom Project Janus. Na operačnom systéme Solaris 10 je spustiteľný binárny kód Solarisu a natívny binárny kód Linuxu vedľa seba bez akýchkoľvek modifikácií, čo umožňuje širší prístup k aplikáciám napísaným pre oba operačné systémy.

ZFS

So 16 miliárd miliárd krát vyššou kapacitou ako aktuálne 64-bitové súborové systémy bude ZFS jediným súborovým systémom odvetvia s prakticky neobmedzenou kapacitou ukladania dát, čo pomôže urobiť zo Solarisu 10 najlepší operačný systém odvetvia na ukladanie dát. ZFS dokáže znížiť administratívne náklady až o 80 percent a poskytuje dokázateľnú integritu dát s 99,99999999999999999-percentnou šancou detekcie a korekcie chyby.

Cryptographic Framework

Na základe odvetvím akceptovaného štandardu PKCS#11 tento nový rámec zlepšuje výkon a centralizuje manažment kryptografických operácií. Vývojári aplikácií môžu písať do spoločného API, čo pomáha organizáciám chrániť svoje investície do aplikácií a pomáha partnerom spoločnosti Sun písať aplikácie bez toho, aby sa museli zaujímať o použitú kryptografickú implementáciu.

Sun Update Connection

Aktualizačná služba, ktorá umožňuje používateľom analyzovať systém a udržiavať ho v aktuálnom stave takmer kdekoľvek. Sun Update Connection poskytuje upozornenia, umožňuje používateľom lokálne aktivity pomocou systémového aktualizátora a vzdialené nasadzovanie pomocou aktualizačného portálu spoločnosti Sun.

AMD64

Solaris 10 je optimalizovaný pre 64-bitové procesory AMD Opteron, čím poskytuje oveľa väčší výkon, ako bol výkon doposiaľ dostupný na nízkonákladových architektúrach, pričom naďalej poskytuje binárnu kompatibilitu medzi 32-bitovými a 64-bitovými prostrediami.

Solaris Application Guarantee

Sun ponúka Solaris Application Guarantee, na základe ktorej zaručuje, že aplikácie písané pre predchádzajúce verzie Solarisu pobežia aj na systéme Solaris 10. Táto záruka rozširuje spätnú kompatibilitu na Solaris 2.6, čím pokrýva sedem rokov verzií operačného systému.

2.2.2. Projekt OpenSolaris

Projekt OpenSolaris je projekt open source, ktorý sprístupňuje zdrojový kód na operačný systém Solaris 10 prakticky každému, za účelom inovácie a vytvárania nových produktov. Budúce verzie operačného systému Solaris budú založené na technológii z projektu OpenSolaris, ktorý je k dispozícii na http://opensolaris.org.

2.2.3. Výhody a nevýhody OS Solaris

Nasledujúce vymenovanie sumarizuje všeobecne uznávané najzávažnejšie výhody a nevýhody operačného systému Solaris.

Výhody

- vysoká stabilita
- vysoká bezpečnosť jadra

- inovatívne vlastnosti
- dostupnosť zdarma
- možnosť dokúpenia podpory
- vhodný najmä pre väčšie spoločnosti, ale v súčasnosti čoraz viac preniká aj do oblasti malých a stredných spoločností
- podporovaný množstvom komerčných aplikácií
- otvorený zdrojový kód (zväčšujúca sa komunita prispievateľov, jednoduchšie vyhľadávanie bezpečnostných zraniteľností a tým rýchlejšie delegovanie bezpečnostných opráv)

Nevýhody

- málo podporovaného hardvéru
- administrátori ho väčšinou nepoznajú
- nedostatočná implementácia príťažlivých používateľských rozhraní
- pretrvávajúce predsudky z minulosti (vhodný len na SUN hardvér, komerčný a pod.)
- otvorený zdrojový kód (jednoduchšie vyhľadávanie bezpečnostných zraniteľností a tým jednoduchšia práca potenciálnych útočníkov)

2.3. Porovnanie vlastností jadier operačných systémov Linux, Solaris a FreeBSD

Toto porovnanie[9] sa zameriava na tri hlavné skúmané podsystémy, ktorými sú:

1. plánovanie procesov
2. správa pamäte
3. súborový systém

Tieto tri subsystémy boli vybrané, pretože sú spoločné pre všetky operačné systémy (nie len UNIXové) a ide o pravdepodobne najjednoduchšie pochopiteľné súčasti operačného systému.

Plánovanie procesov

Základom plánovania v OS Solaris je štruktúra `kthread_t`; vo FreeBSD štruktúra `thread`; a v Linuxe štruktúra `task_struct`. Solaris reprezentuje každý proces pomocou `proc_t`, pričom každé vlákno v rámci procesu má unikátne `kthread_t`. Linux reprezentuje procesy aj vlákna pomocou štruktúry `task_struct`. Proces obsahujúci len jedno vlákno v OS Linux je reprezentovaný len jednou inštanciou štruktúry `task_struct`. Proces obsahujúci len jedno vlákno v OS Solaris má jednu inštanciu štruktúry `proc_t`, jednu `kthread_t` a `klwp_t`. Štruktúra `klwp_t` poskytuje bezpečný priestor pre prepínanie medzi užívateľským módom a kernel-módom. Procesu obsahujúcemu len jedno vlákno vo FreeBSD sú priradené inštancie štruktúr `proc_struct`, a `thread_struct`, a `ksegrp_struct`. `ksegrp` znamená "kernel scheduling entity group." Všetky tri operačné systémy vlastné plánujú vlákna, kde vlákno je reprezentované štruktúrou `kthread_t` v Solarise, `thread` vo FreeBSD, a `task_struct` v Linuxe.

Pánovnie procesov je založené na ich priorite. V Linuxe a FreeBSD má proces tým väčšiu prioritu, čím je jeho numericky reprezentovaná hodnota priority menšia. Čiže sa jedná o inverziu. Proces s prioritou 0 má teda najväčšiu prioritu. V OS Solaris je to opačne, čím väčšia numerická hodnota priority, tým väčšiu prioritu daný proces/vlákno má. V tabuľke 1 sú ukázané hodnoty priorít spomínaných operačných systémov.

Solaris	**Priorita**	**Plánovacia trieda**
	0-59	Time Shared, Interactive, Fixed, Fair Share Scheduler
	60-99	System Class
	100-159	Real-Time
	160-169	Low level Interrupts
Linux	**Priorita**	**Plánovacia trieda**
	0-99	System Threads, Real time (SCHED_FIFO, SCHED_RR)
	100-139	User priorities (SCHED_NORMAL)
FreeBDS	**Priorita**	**Plánovacia trieda**
	0-63	Interrupt
	64-127	Top-half Kernel
	128-159	Real-time user
	160-223	Time-share user
	224-255	Idle user

Tabuľka 1: rozsahy priority procesov a plánovacie triedy v operačných systémoch Solaris, Linux a FreeBSD

Všetky tri operačné systémy uprednostňujú interaktívne vlákna, resp. procesy. Interaktívne procesy bývajú spúšťané s vyššou prioritou, no na kratšie časové intervaly. Solaris, FreeBSD a Linux používajú tzv. per-CPU runqueue, čo znamená, že pre každý procesor je vytvorený osobitný plánovací rad procesov/vlákien. FreeBSD a Linux používajú rady *active* a *expired*. Vlákna sú plánované podľa priority z radu *active.* Keď vlákno vyčerpá jemu pridelený časový interval, je premiestnené do *expired* radu (toto sa deje aj v iných prípadoch). Keď sa rad *active* vyprázdni, jadro ho zamení za rad *expired* a pokračuje v plánovaní. FreeBSD má aj tretí rad označený ako *idle*. Vlákna v tomto rade sú súpšťané len vtedy, keď sú zvyšné dva rady prázdne. Solaris používa tzv. *dispatch queue* pre jednotlivé CPU. Keď vlákno využije jeho časový interval, jadro mu priradí novú prioritu a vráti ho do *dispatch queue*. Rady pre všetky tri operačné systémy majú oddelené zoznamy spustiteľných vlákien pre rôzdne priority. (FreeBSD používa jeden zoznam pre štyri priority, Solaris a Linux používajú oddelené zoznamy pre každú prioritu.)

Linux a FreeBSD používajú arithmetické výpočty založené na pomere časov, kedy vlákno beží a kedy spí (run time versus sleep time) ako meradlo interaktívnosti na správne priradenie priority vláknu. Ani jeden z troch porovnávaných operačných systémov nepoužíva *gang scheduling*. Namiesto plánovania N vlákien, každý operačný systém plánuje len ďalšie vlákno, ktoré bude spustené za aktuálne

bežiacim. Všetky tri systémy majp mechanizmy, ktorými využívajp výhody cyrovnávacej pamäte (cache) a load balancing-u. Pre procesory s podporou systému hyper-threading FreeBSD má mechanizmy na udržanie vlákien na jednom fyzickom profesore. Solaris má podobný mechanizmus, ale je pod kontrolou používateľa a aplikácie a nie je obmedzený len na hyper-vlákna (v Solarise nazývané *processor sets* a vo FreeBSD *processor groups*).

Jeden z veľkých rozdielov medzi Solarisom a ďalšími dvomi OS je možnosť podpory viacerých plánovacích tried[1] (*scheduling classes*) v systéme v rovnakom čase. Všetky tri operačné systémy podporujú Posix `SCHED_FIFO`, `SCHED_RR` a `SCHED_OTHER` (alebo `SCHED_NORMAL`). `SCHED_FIFO` a `SCHED_RR` sú typicky určené pre realtime vlákna[2]. Solaris podporuje triedy s fixnou prioritou a taz. Systémovú triedu (*system class*) pre systémové vlákna. Ďalej sa tam nachádza interaktívna (*interactive*) trieda používaná pre vlákna grafického používateľského rozhrania (typicky okenné aplikácie pod kontrolou X servra). Ďalšou špecialitou solarisu je tzv. *Fair Share Scheduler*[3] určený na podporu správy systémových zdrojov. V tejto práci mu je venovaná samostatná kapitola. Typ plánovania procesov je vo FreeBSD vyberaný pri kompilácii jadra, v Linuxe typ plánovača závisí od verzie jadra.

Správa pamäte a stránkovanie

Každému procesu v OS Solaris prislúcha adresný priestor pozostávajúci z oddelených logických sekcií nazývaných *segmenty*. Segmenty prislúchajúce adresnému priestoru daného procesu sú viditeľné pomocou `pmap`. Solaris rozdeľuje kód pre správy pamäte (memory management code) a dátové štruktúry na platformovo nezávislé a platformovo špecifické oblasti. Tieto sú navzájom prepojené pomocou vrstvy zvanej *hardware address translation*, alebo HAT. FreeBSD popisuje adresné rozsahy jeho procesov pomocou tzv. *vmspace*, ktoré sú rozdelené do logických sektorov nazývaných *regions*. Hardvérovo závislé oblasti zabezpečuje modul *pmap* (physical map), hardvérovo nezávislé oblasti a dátové štruktúry rieši *vmap*. Linux používa pamäťové deskriptory na rozdelenie adresného priestoru procesu na logické sekcie nazývané pamäťové oblasti (memory areas). Linux tiež obsahuje príkaz `pmap` na prehliadanie adresnej oblasti procesu.

Linux oddeľuje strojovo závislé vrstvy os strojovo nezávislých vrstiev na oveľa vyššej úrovni v softvéri. V Solarise a FreeBSD väčšina kódu súvisiaceho napríklad z ošetrovaním page fault sa nachádza v strojovo nezávislej vrstve. V Linuxe sa kód

1 Ďalšie informácie o plánovacích triedach je možné nájsť v manuálovej stránke priocntl(1) OS Solaris.

2 Jadrá Solaris a Linux podporujú preempciu na podporu realtime vlákien.

3 Ďalšie informácie o Fair Share Scheduler je možné nájsť aj v manuálovej stránke FSS(7) OS Solaris.

ošetrujúci page fault nachádza oveľa viac v hardvérovo závislej vrstve. Z toho vyplýva, že Linux môže ošetroať stránkovanie oveľa rýchlejšie, pretože v jadre dochádza k menšej dátovej abstrakcii (vrstveniu). Naopak nevýhodou je, že zmena nižšej vrstvy vyžaduje oveľa viac zásahov do zdrojového kódu. Solaris a FreeBSD izolujú podobné zmeny vo vrstve *HAT*, resp. *pmap*.

Oblasti pamäte sú určované:

- Virtuálnou adresou začiatku oblasti.
- Ich umiestnením v rámci objektu/súboru, ktorý je namapovaný na segment/region/oblasť pamäte.
- Oprávneniami.
- Veľkosťou mapovania.

Mechanizmy v týchto troch operačných systémoch sú veľmi podobné, ale mená dátových štruktúr sú úplne odlišné. Opäť platí, že jadro OS Linux je viac strojovo závislé ako v prípade OS Solaris či FreeBSD.

Stránkovanie

Všetky tri operačné systémy používajú obmeny algoritmu LRU (least recently used) pre uchovávanie, resp. nahrádzanie stránok. Všetky tri obsahujú proces/vlákno realizujúci nahrádzanie stránok. Na FreeBSD sa pravidelne spúšťa démon *vm_pageout* keď sa množstvo voľnej pamäte zmenší. Keď sa množstvo dostupnej pamäte zníži pod určitú hranicu, *vm_pageout* spustií rutinu *vm_pageout_scan,* ktorá prehľadá pamäť za účelom hľadania voľných stránok. Rutina *vm_pageout_scan* vyvolať potrebu asynchrónneho zápisu modifikovaných stránok na disk pred tým, ako ich uvoľní. V systéme beží práve jeden takýto daemon bez ohľadu na počet procesorov. Solaris tiež obsahuje *pageout daemon* ktorý je rovnako periodicky spúšťaný ako odzva na nedostatov voľnej pamäte. Hranice (paging thresholds) pri ktorých sú vyvolávané spomenuté udalosti sú v Solarise automaticky kalibrované počas štartu systému. FreeBSD daemon používa hodnoty, ktoré sú pre väčšinu častí napevno stanovené, prípadne upraviteľné kvôli určeniu spomenutých limitov (paging thresholds). Linux tiež používa algoritmus LRU, ktorý je dynamicky upravovaný počas behu. V Linuxe môže bežať viac daemonov *kswapd* v závislosti od poštu procesorov. Vo všetkých porovnávaných operačných systémoch sú pravidlá platné pre celý systém, sú teda pre všetky procesy rovnaké (global working set policy).

FreeBSD obsahuje niekoľko zreťazených zoznamov stránok na udržiavanie informácií o ich poslednom použití. Rozoznávajú sa stavy stránky: *active*, *inactive*, *cached* a *free*. Stránky sú presúvané medzi týmito zreťazenými zoznamami v závislosti od ich používania. Často používané stránky zostávajú v zozname *active*. Stránky procesu, ktorý sa práve ukončuje môžu byť okamžite premiestnené do zoznamu *free*. FreeBSD môže uložiť celú pamäť procesu na disk (taz. Swapovanie) ak *vm_pageout_scan* nedokáže udržať voľnú pamäť na požadovanej úrovni (napríklad ak systém nedisponuje dostatočne veľkou fyzickou operačnou pamäťou). Ak nastane vážna situácia, pri ktorej systém nedokáže uspokojovať požiadavky na alokáciu pamäte, *vm_pageout_scan* ukončí (kill) procesy, ktoré zaberajú najviac pamäte s najväčším množstvom alokovanej pamäte.

Linux tiež používa rôzne zreťazené zoznamy stránok potrebných pre algoritmus typu LRU. Linux rozdeľuje fyzickú pamäť na tri zóny: jednu pre DMA stránky, druhú pre normálne stránky a tretiu pre dynamicky alokovanú pamäť. Stránky sú presúvané medzi zoznamami: *hot*, *cold* a *free*. Migrácie medzi spomenutými zoznamami je veľmi podobná mechanizmu vo FreeBSD. Často používané stránky sú vedené v zozname *hot*, voľné stránky prechádzajú do zoznamov *cold* a *free*.

Solaris používa zoznamy *free list*, *hashed list* a *vnode page list* na chod jeho variantu algoritmu LRU. Namiesto prehľadávania zoznamov *vnode list* alebo *hash page list* (ktoré sú viac-menej ekvivalentmi zoznamov *active*/*hot* implementovanými v FreeBSD/Linux), Solaris prehľadáva všetky stránky s použitím algoritmu *two-handed clock*[1]. V prípade tejto implementácie dve ruky (podľa terminológie algoritmu) zostávajú navzájom vo fixnej vzdialenosti. Predná ruka nastavuje vek stránky vypnutím referenčného bitu/bitov stránky. Ak žiadny proces nepristúpil (nereferencoval) k stránke, od jej nastavenia prednou rukou, zadná ruka stránku uvoľní a ak bola modifikovaná, asynchrónne ju zapíše na pevný disk.

Všetky tri operačné systémy berú do úvahy NUMA (Non-Uniform Memory Access) spôsob prístupu do pamäte. Rovnako v prípade všetkých troch porovnávaných OS sú vstupno-výstupná vyrovnávacia pamäť (I/O buffer cache) a vyrovnávacia pamäť virtuálnych pamäťových stránok (virtual memory page cache) spojené do jednej systémovej stránkovacej vyrovnávacej pamäte (system page cache). Táto vyrovnávacia pamäť je používaná pre operácie zápisu a čítania zo súborov, rovnako aj pre aplikačné dáta.

1 Algoritmus two-handed clock je popísaný v [14] - Solaris Internals.

Súborový systém

Porovnaniu súborových systémov spomenutých operačných systémov sa v tejto časti nebudem venovať, pretože dostupná literatúra uvádza pri OS Solaris len staršie súborové systémy (najmä UFS). V súčasnosti je už ale presadzovaný súborový systém ZFS, ktorému je v tejto práci venovaná samostatná kapitola.

2.4. Solaris kontajnery

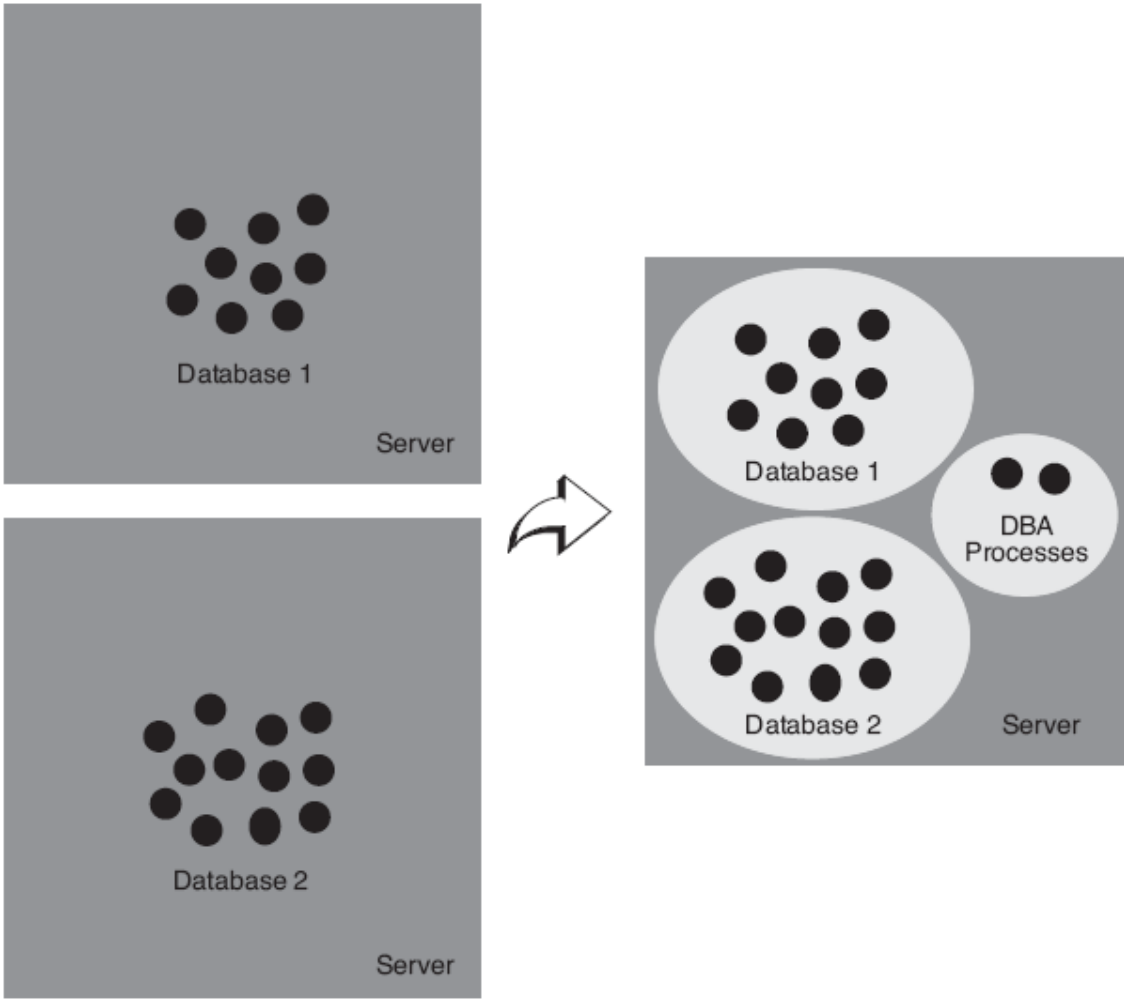
V súčasnosti firmy často navrhujú ich systémy s prídavnou kapacitou na zvládanie občasných najvyšších zaťažení, aby čo najviac zvýšili výnos v období vysokého dopytu. Táto prídavná systémová kapacita ostáva nevyužitá počas obdobia normálnej prevádzky. Počas obdobia zvýšených nárokov, môžu byť zdroje dynamicky prerozdelené medzi dôležité aplikácie. Zdieľanie zdrojov touto cestou vedie k zvýšenému využitiu zdrojov, zmenšuje náklady na riadenie a ceny celkového systému tým, že redukuje počet požadovaných systémov. Pre efektívnu konsolidáciu aplikácií do menšieho počtu fyzických systémov, musí zostať možnosť nezávislého riadenia jednotlivých aplikácií. To si vyžaduje možnosti kontroly využitia zdrojov, izolácie porúch a správy bezpečnosti medzi viacerými aplikáciami na jednom serveri. Inak povedané, vyžaduje si to vytvorenie virtuálnych serverových hraníc vnútri fyzického serveru.

Prvým krokom v tomto smere bolo predstavenie dynamických systémových domén (Dynamic System Domains) na veľkých serveroch SUN. S dynamickými systémovými doménami môže byť server rozdelený do niekoľkých domén, pričom na každej beží vlastná kópia operačného systému Solaris. Domény zabezpečujú hardvérovú izoláciu aplikácií, takže poškodenie v jednej doméne sa neprenesie do iných domén. Doménové hranice môžu byť dynamicky rozdelené aby bola možná ich adaptácia na zmeny systémových zdrojov. Zdroje sa môžu presúvať z jednej domény do inej bez nutnosti reštartovania systému.

2.4.1. Správa úloh

Štandardne, operačný systém Solaris spracúva každú požiadavku na zdroje s rovnakou prioritou. Ak je k dispozícii dostatok zdrojov, žiadosť je akceptovaná a zdroje sú pridelené. Ak požiadavka na zdroj prekračuje mieru celkovej dosiahnuteľnej kapacity, operačný systém Solaris sa prispôsobí obmedzením

prístupu k zdroju. Akcia použitá na obmedzenie prístupu, závisí od typu zdroja. Napríklad, ak by mala požiadavka na čas CPU presiahnuť dosiahnuteľný čas CPU, plánovač procesov zareaguje nastavením priority procesov za účelom zmeny rozdelenia času CPU. Rozvrhovací program operuje s vláknami, pričom nemá žiadny koncept aplikácií. Neuvažuje teda si ich vzájomnú previazanosť a dôležitosť z hľadiska obchodnej perspektívy. Nedvôležitá aplikácia nadmerne vyťažujúca procesor môže diskriminovať iné, oveľa dôležitejšie aplikácie.



Obrázok 3: Kontajnery Solaris zabezpečujú prostredie, ktoré podporuje bezpečné usporiadanie aplikácií do jediného fyzického servera.

Iné zdroje, tak ako celkový počet procesov v systéme, majú pevnú hornú hranicu. Len čo je limit dosiahnutý tento zdroj nemôže byť viac použitý. Prebiehajúci proces, ktorý pokračuje v tvorbe nových procesov, môže zabrániť štartu nových užitočných procesov. Inak ako špecifikovať hornú hranicu na úrovni systému, nie je iná možnosť ako obmedziť počet procesov, ktoré môžu byť vytvorené aplikáciou alebo sadou aplikácií.

Je teda potrebný spôsob kontroly využívania zdrojov založený na akejsi pracovnej náplni (workload – ďalej **úloha**). Úloha je zoskupenie viacerých procesov prislúchajúcich aplikácii alebo skupine aplikácií, ktoré spolu súvisia najmä z hľadiska obchodnej perspektívy. Namiesto riadenia využitia zdrojov na úrovni procesov, je teda možné riadiť využitie zdrojov na úrovni akýchsi pracovných úloh. Toto umožní aplikáciu pravidiel typu „Obchodná aplikácia má garantovaných minimálne 30 percent výpočtových zdrojov" čo môže smerovať k dohodám typu SLA (service level agreement) na úrovni operačného systému. Vlastnosti riadenia zdrojov operačného systému Solaris umožňujú pracovať s úlohami spôsobom:

- obmedzenie prístupov k špecifickým systémovým zdrojom
- poskytnútie zdrojov jednotlivým úlohám na základe preferencií
- vzájomná izolácia úloh

Prvým krokom pri riadení využívania zdrojov úlohami je identifikovanie a klasifikácia komponentov, ako sú procesy z ktorých daná úloha pozostáva. Ďalším krokom je meranie spotreby zdrojov týmito úlohami. Nakoniec môže byť úloha kontrolovaná aplikovaním obmedzení na využívanie pre ňu určených zdrojov. Aplikované obmedzenia vychádzajú z metód definovaných pre pracovnú náplň, založených na obchodných požiadavkách. Vhodným postupom by mohlo byť aplikovanie pravidla garantujúceho dôležitej úlohe určené minimálne množstvo systémových zdrojov, dokonca aj v stave preťaženom systému. Opačnou metódou by mohol byť prístup, kedy úloha dostane prístup k CPU, ak aktuálne žiadne iné úlohy prístup k tomuto zdroju nepožadujú.

2.4.2. Projekty

Prvým krokom v správe systémových zdrojov je určenie úloh (workloads) bežiacich v systéme. Vhodné prístupy zahŕňajú určenie úloh pomocou užívateľského mena alebo názvu procesu. Tieto prístupy sú jednoduché, ale predstavujú problém v prípade, keď viacero inštancií tej istej aplikácie beží v systéme v rámci rôznych úloh, ako napríklad databáza pre obchodnú aplikáciu a databáza pre marketingovú aplikáciu. Pokiaľ databázová aplikácia nepodporuje spúšťanie vlákien pod rôznymi užívateľmi v závislosti od úlohy, nie je použiteľný ani jeden zo spomenutých prístupov. Navyše to neumožňuje zhromaždenie viacerých súvisiacich aplikácií, ako databázové servery, aplikačné servery a webové servery pre obchodné aplikácie v jednom systéme.

Operačný systém Solaris poskytuje možnosť zvanú projekty na určenie úloh (workloads). Projekt slúži ako administratívna značka používaná na zoskupenie súvisiacej práce, do určitej miery vhodne stanovenej správcom systému. Správcovia systému môžu, napríklad vytvoriť jeden projekt pre obchodné aplikácie a iný projekt pre marketingové aplikácie. Umiestnením všetkých procesov súvisiacich s predajnými aplikáciami do predajného projektu a procesov pre marketingové aplikácie do marketingového projektu, môže administrátor rozdeľovať a hlavne riadiť úlohy určitým spôsobom vzhľadom na ich postavenie v obchodnej činnosti.

Používateľ, ktorý je členom viac než jedného projektu môže spúšťať procesy súčasne vo viacerých projektoch, čo umožňuje používateľovi súčasne participovať na viacerých úlohách. Všetky detské procesy spustené istým rodičovským procesom po ňom dedia aj príslušnosť k projektu. Čiže nastavenie príslušnosti štartovacieho skriptu k istému projektu spôsobí spustenie všetkých aplikácií z daného skriptu ako častí daného projektu.

Databáza projektov

Projekty sú definované v databáze projektov. Tá sa môže nachádzať v lokálnom súbore, alebo v adresárovej službe typu NIS či LDAP. Umiestnením databázy projektov do NIS alebo LDAP môžu byť definície projektov zdieľané medzi viacerými systémami. Každá položka (entita) databázy projektov pozostáva z nasledovných polí:

- **name** – meno projektu
- **ID** – unikátny numerický identifikátor projektu
- **comment** – popis projektu
- **user list** – zoznam užívateľov zahrnutých do projektu
- **group list** – zoznam skupín zahrnutých do projektu
- **attributes** – zoznam atribútov projektu (napr. ovládače zdrojov)

Novo nainštalovaný systém vždy obsahuje lokálnu databázu projektov v súbore **`/etc/project`**. Táto obsahuje päť štandardných projektov:

- **system** - systémové procesy a deemony
- **user.root** - všetky procesy bežiace pod užívateľom `root`
- **noproject** - procesy špecifické pre IP quality of service (IPQoS)
- **default** – procesy užívateľov ktorí neprislušia k žiadnemu projektu (zahŕňa všetky projekty)
- **group.staff** – procesy všetkých užívateľov v skupine `staff`

Užívateľ alebo skupina môže byť členom jedného alebo viacerých projektov. Zoznam užívateľov a skupín v databáze projektov určuje, v ktorých projektoch môže užívateľ, alebo skupina užívateľov spúšťať procesy. Tieto zoznamy môžu obsahovať tzv. wildcards ktoré umožňujú flexibilne definovať rozsahy užívateľov. Užívatelia sa (resp. ich procesy) môžu prepnúť do akéhokoľvek projektu, ktorého sú členmi. Pokiaľ sa užívateľ neprepne do iného projektu,. Jeho procesy sú spúšťané v jeho prednastavenom projekte (default project). Zoznam užívateľov a skupín definuje len projekty v ktorých je užívateľovi dovolené spúšťať procesy, nedefinuje však predvolený projekt pre daného používateľa alebo skupiny. Tento je určený systémom počas prihlasovania užívateľa do systému.

2.4.3. Fair Share Scheduler

Beh viacerých úloh v rovnakom systéme môže viesť k situácii kedy jedna úloha obsadí výpočtové zdroje, čím postihne aj ostatné úlohy. Toto môže mať dopad na dôležité úlohy, pretože im nebude poskytnutá dostatočná výpočtová kapacita ku korektnému behu. Je vhodné mať mechanizmus, pomocou ktorého systémoví administrátori môžu uprednostniť prístup k výpočtovým prostriedkom v závislosti od dôležitosti (priority) danej úlohy.

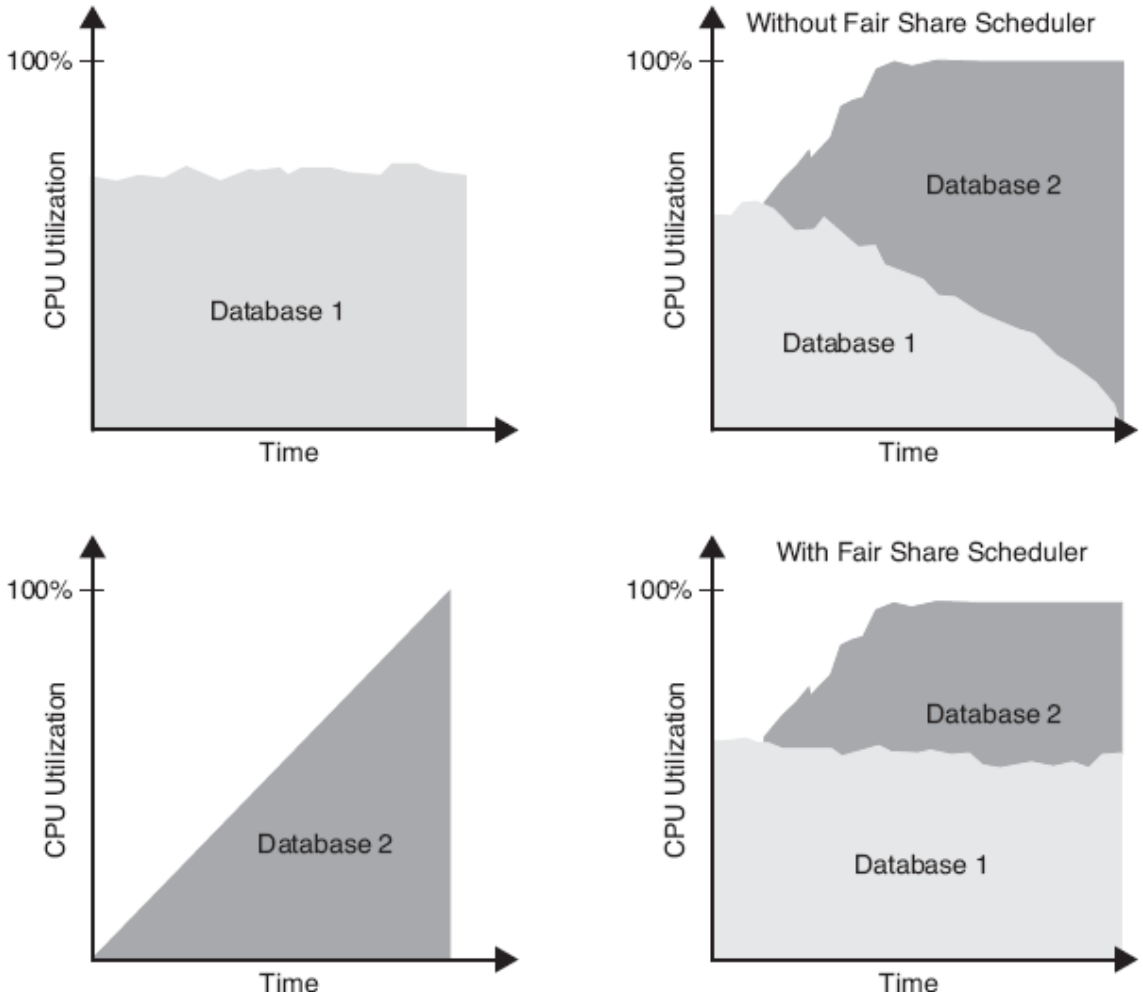
Štandardný plánovač procesov v OS Solaris poskytuje každému vláknu relatívne rovnaké výpočtové zdroje. Ak nemá žiadne informácie o úlohách, štandardný plánovač nemôže uprednostniť pridelenie výpočtových zdrojov v závislosti od dôležitosti úlohy. Solaris ponúka alternatívny plánovač, ktorý dokáže vziať do úvahy priority plánovaných úloh a podľa nich rozdeľovať výpočtové zdroje.

Podiely CPU (SPU Shares)

Fair Share Scheduler (FSS) ovláda prideľovanie výpočtových zdrojov použitím tzv. CPU Shares (ďalej podiely CPU). Dôležitosť úlohy je vyjadrená počtom podielov, ktoré systémový administrátor pridelí projektom reprezentujúcim úlohu. FSS zabezpečuje, že výpočtové zdroje sú rozložené medzi aktívne projekty, ktoré sú založené na počte podielov pridelených každému projektu. (viz. Obrázok 5).

Podiel CPU definuje relatívny nárok výpočtových zdrojov, ktoré sú v systéme k dispozícii pre daný projekt. Je potrebné poznamenať, že podiely CPU nie sú to isté, ako percentuálna hodnota z využitia procesora. Podiely CPU definujú relatívnu dôležitosť projektov s ohľadom na iné projekty. Ak projekt A je napr. dvakrát taký dôležitý ako projekt B, projektu A by malo byť pridelených dvakrát toľko podielov CPU ako projektu B. Informácia o aktuálnom počte pridelených podielov je dosť

irelevantná – 2 podiely pre projekt A versus 1 podiel pre projekt B pripúšťa rovnaký výsledok ako 18 podielov pre projekt A versus 9 podielov pre projekt B. V oboch prípadoch, projekt A má k dispozícii dvakrát väčšie množstvo výpočtových zdrojov ako projekt B. Dôležitosť projektu A vzhľadom na projekt B môže byť zvýšená pridelením viacerých podielov projektu A pričom počet podielov projektu B zostane nezmenený.

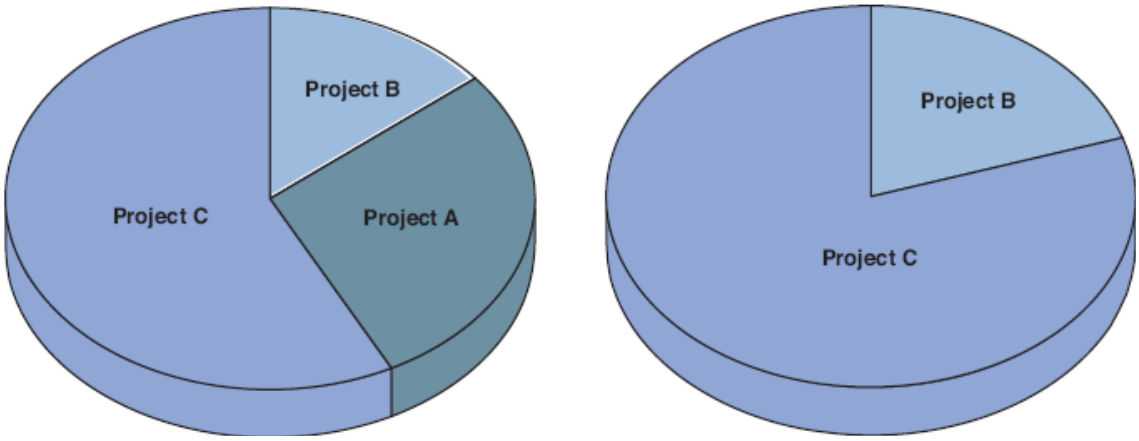


Obrázok 4: Fair Share Scheduler zabezpečuje aby aplikáciam boli pridelené výpočtové zdroje podľa preferencií nastavených administrátorom.

Fair Share Scheduler vypočítava pomer výpočtových zdrojov pridelených projektu deliac počet podielov pre projekt celkovým počtom podielov v aktívnom projekte. Aktívny projekt je projekt s aspoň 1 procesom používajúcim výpočtové zdroje. Podiely pre nečinný projekt, ako sú projekty bez aktívnych procesov, nie sú používané vo výpočtoch. Napríklad, vezmime do úvahy projekty A, B a C s počtom podielov 2, 1 a 4. Ak projekty A, B a C sú aktívne, projekt A je oprávnený využívať 2\7, projekt B na 1\7 a projekt C 4\7 výpočtových zdrojov. Ak projekt A je nečinný, projekt B môže využívať 1/5 výpočtových zdrojov a projekt C 4/5 (viz. Obrázok 5).

Teda aj keď aktuálne množstvo výpočtového výkonu prideleného projektom B a C vzrastá, pomer medzi projektami B a C zostáva nezmenený (1:4).

FSS však limituje použitie výpočtových zdrojov len ak dochádza k súpereniu o ne. Ak je v systéme iba jeden aktívny projekt, môže používať 100% výpočtových zdrojov bez ohľadu na počet podielov CPU, ktoré mu sú pridelené. Nikdy teda nedochádza k neúplnému využitiu výpočtových prostriedkov na úkor spomalenia behu programu. Ak projekt z nejakého dôvodu naplno nevyužíva všetky pridelené výpočtové zdroje, tieto sú prerozdelené medzi ostatné aktívne projekty.



Obrázok 5: Fair Share Scheduler distribuuje výpočtové zdroje medzi aktívne projekty podľa počtu im pridelených podielov CPU.

Konfigurácia podielov CPU

Podiely CPU sú nastavované pomocou **`project.cpu-shares`** ovládača zdrojov (resource control) v databáze projektov. Každému projektu môže byť pridelená hodnota **`project.cpu-shares`** ovládača zdrojov. Projektom bez ovládača zdrojov býva systémom pridelený jeden podiel CPU. Systémový projekt je používaný pre všetky systémové procesy a démonov a je špeciálny v tom, že má neobmedzený počet podielov CPU. Projekty s prideleným nulovým počtom podielov môžu pracovať iba vtedy, ak iné projekty s nenulovými podielmi nie sú aktívne.

Užívatelia môžu byť členmi viacerých projektov a použitie CPU je kontrolované počtom podielov projektu, v ktorom užívateľ pracuje. Dôsledkom tohto, užívateľ môže byť oprávnený použiť rozličné množstvá výpočtových zdrojov v rovnakom čase. Jeden proces ale môže byť súčasne priradený iba jednému projektu, teda mať rozličné množstvá výpočtových zdrojov v rovnakom čase znamená, že procesy, ktoré užívateľ vlastní prislúchajú rozličným projektom.

Na aplikáciu limitovania výpočtového výkonu na jediného užívateľa je potrebné vytvoriť projekt s vhodným počtom podielov CPU, ktorý obsahuje iba daného užívateľa. Tento projekt by mal byť predvolený pre daného užívateľa a užívateľ by nemal byť členom iných projektov.

CPU shares môžu byť nastavené dynamicky pomocou príkazu **`prctl`**. Tieto zmeny sú platné až do reštartu systému. Aby sa zmeny stali zachovali aj po reštarte systému, je potrebné im priradiť pevnú hodnotu v **`project.cpu-shares`** databáze projektov.

2.4.4. Ovládače zdrojov

Množstvo zdrojov priradených úlohám môže byť kontrolované aplikovaním obmedzenia na použitie zdrojov. Tieto hranice môžu byť použité na ochranu úlohy pred pohltením určitého zdroja a následným obmedzovaním iných úloh. Solaris poskytuje možnosť kontroly zdrojov na implementáciu možnosti prevencie pohltenie zdrojov. Táto vlastnosť je rozšírením tradičných vlastností OS UNIX obmedzenie zdroja (*rlimit*). Funkcionalita *rlimit* môže byť použitá na nastavenie obmedzení a limitov využívania výpočtových zdrojov, ako napríklad maximálny procesorový čas, maximálna veľkosť súborov a pod. Avšak, keďže *rlimit* je založený na procesoch, jeho použitie na obmedzenie *workloads* je limitované. Možnosť kontroly zdrojov v OS Solaris rozširuje limity založené na procesoch pridaním kontroly zdrojov na úrovni úloh a projektov. Počet obmedzení systémových zdrojov, ktoré môžu byť nastavené je rozšírený, aby poskytol administrátorom viac kontroly nad využívaním zdrojov procesmi, úlohami a projektmi v systéme.

Správa ovládačov zdrojov

Ovládače zdrojov sú konfigurované pomocou databázy projektov. Posledné pole entity projektu je použitá na nastavenie ovládača zdrojov. Ovládač zdrojov je v entite projektu reprezentovaná dvojicou meno - hodnota. Meno znamená typ limitu, kým hodnota je zoznam atribútov pre ovládač. Projekt môže mať nastavených viac ovládačov zdrojov, ktoré sú navzájom oddelené bodkočiarkami. Zoznam atribútov pre ovládač zdrojov pozostáva z privilegovanej úrovne, hranice a akcie.

Privilegovaná úroveň určuje, ktorý užívateľ môže meniť počiatočnú hodnotu. Rozoznávajú sa tri privilegované úrovne:

- **základná -** vlastník vyvolávajúceho procesu môže meniť medznú hodnotu

- **privilegovaná -** iba privilegovaní (superuser) užívatelia môže meniť medznú hodnotu
- **systémová - medzná hodnota** je nemenná počas celého behu OS

Každý ovládač zdrojov má minimálne systémovú hodnotu, ktorá reprezentuje koľko zdrojov je aktuálna implementácia operačného systému schopná poskytnúť. Ovládač zdrojov môže mať najviac jednu základnú hodnotu a ľubovoľné množstvo privilegovaných hodnôt.

Akcia definuje kroky, ktoré majú byť vykonané po prekročení medznej hodnoty. Možné sú 3 akcie:

- **deny** (odmietnutie) - odmietnuť požiadavku na zdroj pri množstve, ktoré je väčšie ako medzná hodnota
- **signal** - poslať označený signál procesu prevyšujúcemu medznú hodnotu
- **none** (žiadna) - nevykonať žiadnu akciu

Dostupné ovládače zdrojov

Nasledujúca tabuľka identifiku ovládače zdrojov dostupné v OS Solaris 10.

Ovládač zdroja	Popis
`process.max-port-events`	Maximálne dovolené množstvo udalostí na port udalostí (event port).
`process.max-msg-messages`	Maximálne množstvo správ v rade správ (message queue).
`process.max-msg-qbytes`	Maximálne množstvo bajtov správ v rade správ.
`process.max-sem-ops`	Maximálne množstvo semafórových operácií na jedno volanie semfóru.
`process.max-sem-nsems`	Maximálne množstvo semafórov na jednu sadu semafórov (semaphore set).
`process.max-address-space`	Maximálne množstvo adresného priestoru pre jeden proces.
`process.max-file-descriptor`	Maximálny index deskriptora súborov pre daný proces.
`process.max-core-size`	Maximálna veľkosť core súboru, ktorý môže daný proces vytvoriť.
`process.max-stack-size`	Maximálna veľkosť stack segmentu pre daný proces.
`process.max-data-size`	Maximálne množstvo heap pamäte dostupnej pre daný proces.

process.max-file-size	Maximálny offset na ktorý môže daný proces zapisovať.
process.max-cpu-time	Maximálny procesorový čas pre daný proces.
task.max-cpu-time	Maximálny procesorový čas pre procesy danej úlohy.
task.max-lwps	Maximálne množstvo LWP súčasne dostupných procesom danej úlohy.
project.max-contracts	Maximálne množstvo kontraktov dovolené v rámci projektu.
project.max-device-locked-memory	Maximálne celkové množstvo zamknutej pamäte pre daný projekt.
project.max-port-ids	Maximálne množstvo portov udalostí (event ports).
project.max-shm-memory	Maximálne celkové množstvo zdieľanej pamäte pre daný projekt.
project.max-shm-ids	Maximálne množstvo ID zdieľanej pamäte pre daný projekt.
project.max-msg-ids	Maximálne množstvo ID radu správ (message queue Ids) pre daný projekt.
project.max-sem-ids	Maximálne množstvo ID semafórov pre daný projekt.
project.max-crypto-memory	Celkové množstvo pamäte jadra, ktoré môže byť použité hardvérovým akcelerátorom šifrovania pre libpkcs11.
project.max-tasks	Maximálne množstvo úloh pre daný projekt.
project.max-lwps	Maximálne množstvo LWP súčasne dostupných pre daný projekt.
project.cpu-shares	Počet podielov CPU (CPU shares) daného projektu.
zone.max-lwps	Maximálne množstvo LWP súčasne dostupných procesom danej zóny.
zone.cpu-shares	Množstvo podielov CPU pridelených zóne Fair Share Scheduller-om.

Ďalšou zaujímavou vlastnosťou je dynamické prideľovanie procesorov alebo Dynamic resource polls. Viac informácií o ňom je možné nájsť v [7] od strany 39. Statické prideľovanie procesorov je vysvetlené na strane 9 (partitioning) toho istého dokumentu.

2.5. Solaris zóny

Solaris zóny (Solaris Zones) poskytujú prostriedky na vytváranie jedného alebo viacerých virtuálnych prostredí ne jednej inštancii operačného systému, pričom izolujú aplikácie od detailov súvisiacich z hostiteľským operačným systémom. Aplikácie teda nemôžu vidieť, monitorovať ani nijak ovplyvňovať beh iných zón. Zóny poskytujú nasledovné vlastnosti:

- Bezpečnosť
- Izolácia
- Virtualizácia
- Granularita
- Transparentnosť

Bezpečnosť

Sieťové služby uzavreté v zóne obmedzujú dopady na celý systém, ku ktorým by došlo v prípade napadnutia systému zneužitím zraniteľností v daných službách. Útočník, ktorý zneužije zraniteľnosť v danej službe môže teda spôsobiť len obmedzené škody. Akcie ktoré je možné vykonávať v rámci zóny sú obmedzené na podskupinu tých, ktoré je možné vykonávať v normálnom systéme. Napríklad nie je dovolené nahrávať vlastné moduly jadra, pristupovať k pamäti jadra, prípadne vytvárať vlastné zariadenia vo vnútri zóny.

Izolácia

Aplikácie požadujúce exkluzívny prístup ku spoločným zdrojom ako napríklad špecifickým užívateľským menám alebo sieťovým portom môžu s použitím zón bežať na spoločnom hardvéri. Každá zóna má svoj vlastný menný priestor úplne separovaný od ostatných zón. Užívatelia zóny nedokážu monitorovať iné zóny, napríklad nedokážu sledovať sieťovú prevádzku inej zóny aj keď prechádza cez to isté fyzické sieťové rozhranie ako sieťová prevádzka danej zóny. Rovnako nedokážu sledovať procesy v rámci inej zóny.

Virtualizácia

Zóny vytvárajúce virtuálne prostredie pre aplikácie vyrovnávajú rozdiely medzi hardvérovými konfiguráciami serverov a teda aplikácia, ktorá v takejto zóne beží nevníma pri výmene hardvéru žiadny špeciálny (nie výkonový a kapacitný) rozdiel.

Tento fakt uľahčuje distribúciu aplikácií na rôzne fyzické servery, rovnako aj premiestňovanie aplikácií medzi nimi.

Granularita

Keďže zóny sú implementované na úrovni softvéru, ich granularita nie je nijak limitovaná obmedzeniami na úrovni hardvéru. Umožňujú teda aj tzv. sub-CPU granularitu. Nevyžadujú vyhradené výpočtové zdroje, I/O zariadenia ako napríklad. sieťové adaptéry alebo pevné disky, ale ani operačnú pamäť. Následkom spomenutých faktov teda je, že aj základný jednoprocesorový počítač môže byť hostiteľským strojom pre množstvo zón.

Transparentnosť

Prostredie prezentované aplikáciám v rámci zóny je skoro totožné s štandardným prostredím OS Solaris. Na bez aplikácií v rámci zóny nie sú potrebné žiadne špeciálne API. Aplikácie nemusia byť na beh v zóne žiadnym spôsobom portované. Existujú ale niektoré bezpečnostné obmedzenia, ktoré ale ovplyvnia hlavne aplikácie, ktoré vykonávajú privilegované operácie alebo pristupujú priamo k fyzickým zariadeniam.

2.5.1. Základné informácie o zónach

Globálna zóna zahrňuje celý systém a je porovnateľná s bežnou inštanciou OS Solaris. Má prístup k fyzickým zariadeniam a môže vidieť a kontrolovať všetky procesy. Administrátor globálnej zóny môže kontrolovať systém ako celok. Globálna zóna existuje vždy, aj keď nie sú nastavené žiadne iné zóny. Vo vnútri globálnej zóny sa nachádzajú **lokálne zóny**. Tieto sú izolované od fyzického hardvéru **vrstvou virtuálnej platformy** (virtual platform layer). Táto vrstva poskytuje zóne virtuálne sieťové rozhranie, jeden alebo viac súborových systémov a virtuálnu konzolu.

Aj keď virtuálne sieťové rozhrania môžu byť namapované na skutočné sieťové rozhrania, aplikáciám v rôznych zónach je zabránené aby videli sieťovú prevádzku iných zón. Každá zóna má svoje vlastné procesné prostredie a prevádzkuje jej vlastnú sadu základných služieb vrátane `inetd`, `syslogd`, `rpcbind`, and iných. Aplikácie bežiace v zóne nie sú schopné vidieť aplikácie bežiace v iných zónach kvôli ich vlastným privátnym procesným prostrediam (private process environment).

Zóny sú naviazané na im nastavený adresárový podstrom a nemôžu pristupovať na iné miesta v súborovom systéme. Všetky zóny bežia pod tou istou inštanciou operačného systému a teda aj pod jeho rovnakou verziou.

Vrstva virtuálnej platformy nie je emulačnou vrstvou ktorá prekladá požiadavky zo zóny do inej formy prípadne ich spúšťa v mene zóny. Úlohou vrstvy virtuálnej platformy je vytvoriť inštancie virtuálnych zdrojov a pripojiť ich k zóne. Napríklad v prípade **virtuálneho sieťového rozhrania** vrstva virtuálnej platformy vytvorí logické rozhranie namapované na fyzickom rozhraní špecifikovanom v konfigurácii zóny. IP adresa logického sieťového rozhrania a je teda dostupná zo zóny.

Jeden z atribútov logického rozhrania je zóna, v ktorej je nastavené. Jadro používa tento atribút na virtualizáciu sieťového rozhrania tým, že preposiela pakety do príslušnej zóny v závislosti od nastavených atribútov. Zóna teda vidí len pakety, ktoré sú určené pre jej vlastné logické rozhranie. Broadcast-ové a multicast-ové pakety sú preposielané do všetkých zón.

Virtualizácia súborového systému v rámci zóny je dosiahnutá pomocou obmedzeného prístupu mimo stanoveného koreňového (root) adresára podobne ako v prípade `chroot`. Procesy v rámci zóny sú obmedzené na súbory a súborové systémy, ku ktorým môže byť pristupované v rámci obmedzeného (nastaveného) koreňového adresára. Oproti `chroot`, zónu nie je možné opustiť. Vrstva virtuálnej platformy je zodpovedná za vytvorenie stanoveného koreňového adresára a jeho pripojenie (mount) na miesto definované v konfigurácii zóny.

Izolácia procesov je realizovaná pridaním odkazu na zónu do parametrov procesu. Jadro bolo rozšírené tak, aby používalo identifikátor zóny na zabránenie viditeľnosti procesu iným procesom mimo danej zóny. Procesom v danej zóne sú viditeľné len procesy s rovnakým identifikátorom zóny. Toto rozlišovanie je vykonávané na úrovni jadra a teda nie je súčasťou nástrojov ako `ps` alebo `kill`.

Tak, ako identifikátor zóny je súčasťou parametrov, identifikátory užívateľov (UID) sú tiež virtualizované v rámci zón. Každá zóna má jej vlastný zoznam užívateľských identifikátorov. Následkom spomenutého užívatelia s totožným UID v rôznych zónach sú de facto odlišní užívatelia. Kvôli **virtualizácii užívateľských kont** existuje pre každú zónu unikátne úložisko hesiel.

Uvedenie **systému najmenších privilégií** (Least Privilege) do OS Solaris prináša jemnejšiu sadu privilégií ako náhradu konceptu neobmedzeného užívateľa *root*. Kým v štandardnom koncepte prístupu k privilegovaným operáciám sa vykonávala len kontrola toho, či sa užívateľské ID rovná nule, v prípade OS Solaris jadro kontroluje

špecifické privilégiá potrebné na vykonávanie konkrétnych privilegovaných operácií. Príkladom môže operácia mount, na vykonanie ktorej sú potrebné pre užívateľa root špecifické privilégiá. Obmedzením privilégií užívateľa root v rámci lokálnej zóny na bezpečnú úroveň, užívateľ root v lokálnej zóne môže mať stále dostatočné práva na riadenie zóny aj keď nemá možnosť zasahovať do systému ako celku napríklad reštartovaním fyzického servera. Ale obmedzenie privilégií samé o sebe nepostačuje na zabezpečenie dostatočnej izolácie. Privilégiá len obmedzujú operácie, ktoré môžu byť vykonané, ale nekontrolujú objekty, nad ktorými môžu byť dané operácie vykonávané. Práve tento jav je dosiahnutý izoláciou na úrovni zón.

Je teda možné delegovať práva na administráciu lokálnej zóny užívateľom tým, že sa im pridelí prístup na užívateľa root v rámci danej lokálnej zóny. Keďže užívateľ s UID 0 danej zóny je odlišný od užívateľa s rovnakým UID v inej zóne, tento nemôže ovplyvniť žiadnu inú zónu. Užívateľ root v rámci globálnej zóny by ale mal zostať pozorne chránený, pretože mu stále zostáva možnosť kontroly nad celým systémom a prístupu do všetkých prítomných zón.

2.5.2. Rozdiel medzi kontajnermi a zónami

Solaris kontajnery nie sú ekvivalentné Solaris zónam. Technológia zón môže byť použitá na vytváranie kontajnerov s istými charakteristikami ako napríklad izolácia poskytovaná virtualizačným prostredím OS Solaris. Je ale možné vytvoriť aj iný typ kontajneru a to použitím technológie riadenia/zdieľania zdrojov (Resource Pools) ak tú potrebné práve také charakteristiky, aké táto technológia poskytuje. Takže pokiaľ zóna je istým druhom kontajnera, kontajner nemusí byť druh zóny.

2.5.3. Správa zdrojov

V rámci zón sú k dispozícii štandardné funkcie na správu zdrojov ako ovládače zdrojov, projekty a iné. Keďže projekt je navyše virtualizovaný v rámci zóny, každá zóna má svoju vlastnú databázy projektov. Toto umožňuje administrátorovi lokálnej zóny konfigurovať projekty a ovládače zdrojov v rámci danej zóny.

Fair Share Scheduler a Zóny

Na zabránenie stavu, kedy lokálna zóna zahltí celý systém administrátor globálnej zóny môže definovať spôsob rozdeľovania zdrojov medzi zóny. Ovládač zdroja

`zone.cpu-shares` obmedzuje množstvo výpočtových zdrojov ktoré môžu byť pridelené zóne rovnakým spôsobom, ako to *Fair Share Scheduler* realizuje na úrovni projektov. Zóna s vyššou hodnotou `zone.cpu-shares` môže využívať väčšie množstvo výpočtových zdrojov ako zóna s nižšou hodnotou tohto parametra. Parameter (ovládač zdroja) `zone.cpu-shares` je nastavovaný pomocou príkazu `zonecfg`. Toto ale vyžaduje aby bol *Fair Share Scheduler* nastavený ako hlavný plánovač v systéme alebo aby bol danej zóne pridelený istý rozsah procesorov a bol pre ňu nastavený *FSS* ako hlavný plánovač.

V kombinácii s plánovačom *Fair Share Scheduler* na úrovni zóny nastáva prípad kedy je výpočtová kapacita prideľovaná na dvoch úrovniach. Na prvej úrovni je nastavenie `zone.cpu-shares` na úrovni globálnej zóny, kde je určené množstvo výpočtových zdrojov pridelených zóne ako celku. Na druhej úrovni (na úrovni lokálnej zóny) je množstvo výpočtových zdrojov rozdelené medzi aktívne projekty v závislosti od nastavenia `project.cpu-shares` definovaného administrátorom lokálnej zóny.

Ovládače zdrojov

Vo vnútri lokálnej zóny sú k dispozícii všetky štandardné ovládače zdrojov a môžu byť použité administrátorom lokálnej zóny na riadenie prideľovania zdrojov v danej zóne. Administrátor globálnej zóny môže ale obmedziť množstvo tzv. lightweight procesov (LWP) ktoré môžu byť vytvorené zónou. Toto množstvo je možné nastaviť parametrom (ovládačom zdrojov) `zone.max-lwps`.

2.6. Balíčkovací systém OS Solaris

Štandardný balíčkovací systém OS Solaris je veľmi jednoduchý – umožňuje len pridanie, prípadne odstránenie jednotlivých balíčkov. Vykonáva síce kontrolu závislostí, no sám nezabezpečuje ich ošetrovanie. V prípade, že nie sú nainštalované požadované balíčky, systém vyhlási chybu, ale sám dané balíčky nenainštaluje. Rovnako neponúka možnosť online získavania balíčkov z internetu.

Zjavné nedostatky tohto systému sa pokúsila kompenzovať komunita Blastwave (www.blastwave.org), ktorá inšpirovaná systémom *apt-get* z Debian Linux vytvoria nástroj *pkg_get* s podobnou funkcionalitou. Tento nástroj spĺňa základné požiadavky systémových administrátorov, no za zmienku stojí, že nemá napríklad doriešené ošetrovanie závislostí v prípade odstraňovania balíčkov. Prehľadnejšie

porovnanie balíčkovacieho systému SUN-u a ďalších systémov sa nachádza v ďalšej podkapitole.

Všetky systémy však používajú rovnaký formát balíčkov, ktorý je vysvetlený v nasledujúcich podkapitolách.

Na záver sú spomenuté vyhliadky na úplne nový balíčkovací systém, ktorý je v súčasnosti v štádiu príprav pre verziu OS Solaris 11.

2.6.1. Princíp

Balíčkovací systém OS Solaris je založený na balíčkoch – skomprimovaných súboroch, ktoré obsahujú hlavne:

- súbory samotnej inštalovanej aplikácie
- základné informácie o samotnom balíčku
- informácie o umiestneniach a parametroch inštalovaných súborov
- informácie o závislostiach
- post-inštalačné a pre-inštalačné skripty

Súbor pkginfo

Súbor pkginfo je v podstate veľmi jednoduchý súbor popisujúci balík ako taký. Jeho syntax sa skladá zo súboru premenných a ich hodnôt [19].

- PKG - Názov balíka (napr SUNWlibC)
- NAME - Názov, zvyčajne sa zhoduje s DESC
- ARCH - Architektúra, väčšinou i386 alebo SPARC
- VERSION - Verzia
- CATEGORY - Kategória (povinný parameter), najčastejšie používanou kategóriou je *application*
- VENDOR – Autor softvéru
- EMAIL – E-mail adresa
- DESC - Popis, zvyčajne sa zhoduje s NAME
- PSTAMP - Značka autora balíčka
- BASEDIR - prefix – koreňový adresár pre adresárovú štruktúru balíčka

- CLASSES - možnosť profilovať balík na jednotlivé triedy, ktoré majú postupnosť pri inštalácii

Súbor prototype

Tento súbor obsahuje informácie o aplikačných súboroch, adresároch a ich umiestnení. Prvý riadok obsahuje umiestnenie súboru pkginfo. Príklad súboru prototype je:

```
i pkginfo=./pkginfo
f none booklist.txt 0644 bin bin
d none books 0755 bin bin
d none books/ABOUTDOC 0755 bin bin
d none books/ABOUTDOC/ebt 0755 bin bin
f none books/ABOUTDOC/ebt/ABOUTDOC.dat 0644 bin bin
f none books/ABOUTDOC/ebt/ABOUTDOC.edr 0644 bin bin
f none books/ABOUTDOC/ebt/ABOUTDOC.tag 0644 bin bin
s none books/ABOUTDOC/ebt/search.tdr=./toc.tdr
f none books/ABOUTDOC/ebt/toc.tdr 0644 bin bin
d none books/ABOUTDOC/index 0755 bin bin
f none books/ABOUTDOC/index/index.dat 0644 bin bin
f none books/ABOUTDOC/index/vocab.dat 0644 bin bin
f none books/ABOUTDOC/styles.ent 0644 bin bin
```

Závislosti

Závislosti balíčkov sú zaznamenané v súbore depend. Príklad súboru depend balíčka SUNWsshu je:

```
P SUNWcar       Core Architecture, (Root)
P SUNWcakr      Core Solaris Kernel Architecture (Root)
P SUNWkvm       Core Architecture, (Kvm)
P SUNWcsr       Core Solaris, (Root)
P SUNWckr       Core Solaris Kernel (Root)
P SUNWcnetr     Core Solaris Network Infrastructure (Root)
P SUNWcsu       Core Solaris, (Usr)
```

```
P SUNWcsl        Core Solaris Libraries
P SUNWcsd        Core Solaris Devices
```

Poznámka: parameter P (prerequisite for installation) určuje, že sa jedná o vyžadovaný balíček.

2.6.2. Zdroje balíčkov

V súčasnosti existujú pre operačný systém Solaris 10 tri hlavné zdroje balíčkov:

- Spoločnosť SUN Misrosystems
- komunita SunFreeware
- komunita Blastwave

Nasledujúca tabuľka uvádza porovnanie spomínaných zdrojov balíčkov:

Balíčkovací systém	**SUN balíčky**	**SunFreeware**	**blastwave**
Predpona balíčkov	SUNW	SFW	CSW
Koreňový inštalačný adresár	/	/usr/sfw	/opt/csw
Zdroje	Inštalačné médiá výrobcu	HTTP/FTP mirrory projektu Sunfreeware	HTTP/FTP mirrory projektu Blastwave
Inštalácia balíčkov	Manuálna inštalácia jednotlivých balíčkov nástrojom pkgadd	Automatizovaná inštalácia nástrojom `pkg-get`	Automatizovaná inštalácia nástrojom `pkg-get`
Odstránenie balíčkov	Manuálne odstránenie jednotlivých balíčkov nástrojom pkgrm	Odstránenie nástrojom `pkg-get` ale bez kontroly závislostí	Odstránenie nástrojom `pkg-get` ale bez kontroly závislostí
Aktualizácia balíčkov	Manuálne odstránenie starého a inštalácia nového balíčka nástrojmi pkgrm a pkgadd	Automatizovaná aktualizácia nástrojom`pkg-get`	Automatizovaná aktualizácia nástrojom `pkg-get`

Podpora softvéru výrobcom

Okrem vlastných softvérových produktov garantuje Spoločnosť SUN Microsystem podporu aj vybraným Open source aplikáciám. Príklady plne podporovaných aplikácií sú:

- Apache
- Apache2
- bind
- ppp
- Samba
- sendmail
- SER (SIP Proxy Server)
- wu-ftpd
- Zebra
- Perl
- Secure Shell
- traceroute

Okrem vymenovaných aplikácií poskytuje spoločnosť SUN Microsystems aj tzv. Moderated podporu, ktorá je realizovaná vo forme poskytovania aktualizácií, ktoré sú dodávané výrobcami jednotlivých aplikácií softvéru. Presný zoznam podporovaného a čiastočne podporovaného softvéru je možné nájsť na stránke:

http://www.sun.com/software/solaris/freeware/

2.6.3. Aktualizácie

Na aktualizáciu softvérových balíčkov sa používajú spomínané balíčkovacie systémy. OS Solaris má však úplne separátny systém záplat nazývaný *Solarich Patch Management*, ktorý je určený hlavne pre systémové komponenty.

2.6.4. Balíčkovací systém projektu Indiana

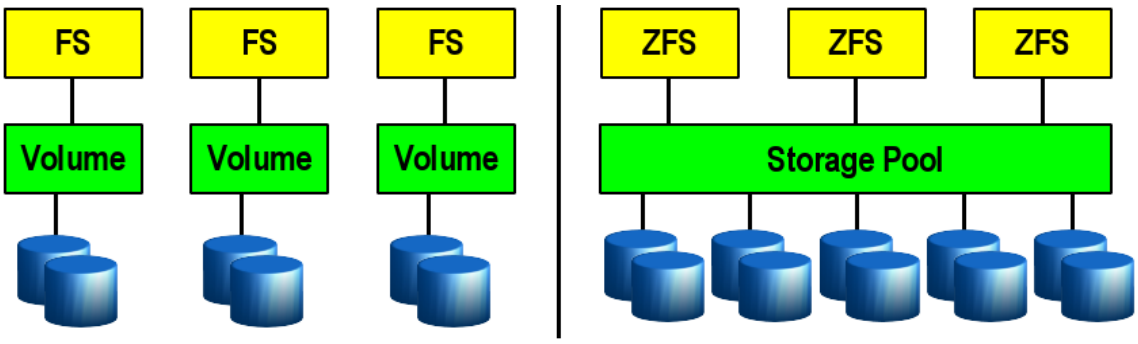
V rámci projektu *Indiana*, ktorý je jedným z projektov realizovaných v rámci *OpenSolaris* je *Image Packaging System* (IPS). Tento je v súčasnosti v štádiu vývoja a kladie si za cieľ vytvorenie multiplatformového univerzálneho balíčkovacieho systému. Niektoré z jeho avizovaných vlastností by mali byť:

- Univerzálnosť (podpora operačných systémov Solaris, Linux, Windows, Mac OS a iných)
- Možnosť inštalovať balíčky aj pre ne-administrátorských používateľov

- Skupinovanie balíčkov
- Jednoduché vytváranie a uverejňovanie vlastných balíčkov
- Možnosť návratu do stavu pred aktualizáciou v prípade jej neúspechu
- Možnosť správy balíčkov na viacerých serveroch súčasne
- Naviazanie zadefinovaných akcií na operácie pridania, resp. odstránenia balíčka (tzv. trigger)
- Závislosti na súboroch, ktoré nie sú súčasťou balíčkov

2.7. Zettabyte File System

ZFS na rozdiel od bežne známych súborových systémov pridáva do jadra operačného systému medzivrstvu, ktorá zabezpečuje mapovanie logických sektorov súborového systému na fyzické umiestnenie dát na dátových médiách a to nie len na úrovni jedného fyzického média (najčastejšie pevného disku), ale am medzi rôznymi diskami.



Obrázok 6: Porovnanie obvyklého prístupu k súborovým systémom (vľavo) a prístupu systému ZFS.

Stručný opis vlastností

Je to 128-bitový súborový systém s v súčasnosti prakticky neprekročiteľnými limitmi na počet a veľkosti súborov.

ZFS priamo podporuje použitie viacerých diskov, RAID 1 a RAID-Z polí, na ktorých môže dynamicky umiestňovať viacero operačných systémov a nevyžaduje dodatočný volume manager pre zabezpečenie tejto funkčnosti.

Podporuje premenlivú veľkosť bloku pre efektívnejšie využitie miesta podľa typu ukladaných súborov.

Zároveň podporuje možnosť verzionovania, kedy pri zmene dát neprepisuje na disku staré bloky ale vytvára ich kópiu. To umožňuje uchovávať všetky verzie menených súborov a vytvárať tzv. snapshoty, kópiu stavu súborového systému vo zvolenom čase v minulosti.

Tieto aj ďalšie zaujímavé vlastnosti sú podrobnejšie vysvetlené v ďalších častiach tejto podkapitoly.

Podpora v iných operačných systémoch

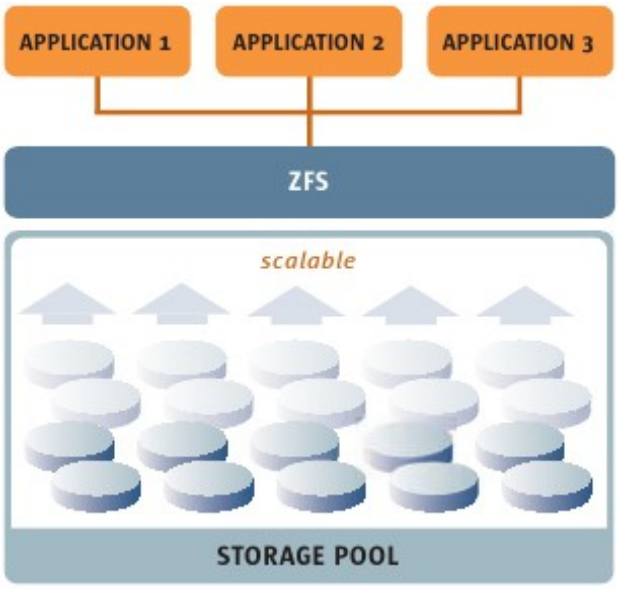
Zettabyte File System bude štandardným súborovým systémom v pripravovanej novej verzii operačného systému Mac OS X 10.5 Leopard [21]. Okrem spomenutého už existuje implementácia tohto súborového systému v operačných systémoch FreeBSD aj Linux (prostredníctvom modulu FUSE (Filesystem in Userspace)).

2.7.1. Storage pools

ZFS je založený na virtuálnych dátových oblastiach (*pools*) nazývaných *zpools*. Jeden zpool, ktorý tvorí základ systému ZFS môže pozostávať z mnohých fyzických zariadení. Zpool pozostáva z virtuálnych zariadení (*vdevs*), ktoré sú zas zložené z blokových zariadení ako súbory, partície, alebo skutočné fyzické zariadenia. Blokové zariadenia v rámci *vdev* môžu byť nastavené rôznymi spôsobmi v závislosti od potrieb a dostupného miesta[22]. Možnosti nastavenia sú napríklad:

- bez redundancia (podobne ako RAID 0)
- mirror (RAID 1) dvoch a viacerých zariadení
- RAID-Z skupina dvoch a viacerých zariadení
- RAID-Z2 skupina troch a viacerých zariadení

Dátová kapacita všetkých virtuálnych zariadení *vdev* je dostupná všetkým súborovým systémom v rámci daného *zpool*.



Obrázok 7: Mnoho fyzických diskov spojených do jedného storage pool-u. Viac aplikácií tak môže pristupovať k spojenej kapacite mnohých fyzických zariadení.

Na jeden súborový systém (partícia[1] - podmnožina *zpool*) môže byť nastavená **kvóta** určujúca maximálnu kapacitu, ktorú môže daná partícia zaberať z jeho *zpool*. ZFS nepodporuje kvóty pre užívateľov, odporúčaným riešením tu je vytvorenie osobitnej partície pre každého používateľa.

Systém **rezervácií** naopak umožňuje danej partícii zaručiť, že určenú kapacitu bude mať určite k dispozícii.

Ako je vidieť, veľkosti aj umiestnenia partícií sa sú v ZFS úplne dynamické.

1 Anglické dokumenty popisujúce súborový systém ZFS nepoznajú pojem partícia (partition). Namiesto neho používajú pojem filesystem. V slovenskom preklade ale uvádzam pojem partícia pre lepšie pochopenie problematiky.

2.7.2. Teoretické limity

ZFS je 128-bitový súborový systém, je navrhnutý tak, aby žiadne z jeho limitov neboli prakticky dosiahnuteľné v súčasnosti ani dohľadnej budúcnosti. Niektoré jeho teoretické limity sú[22]:

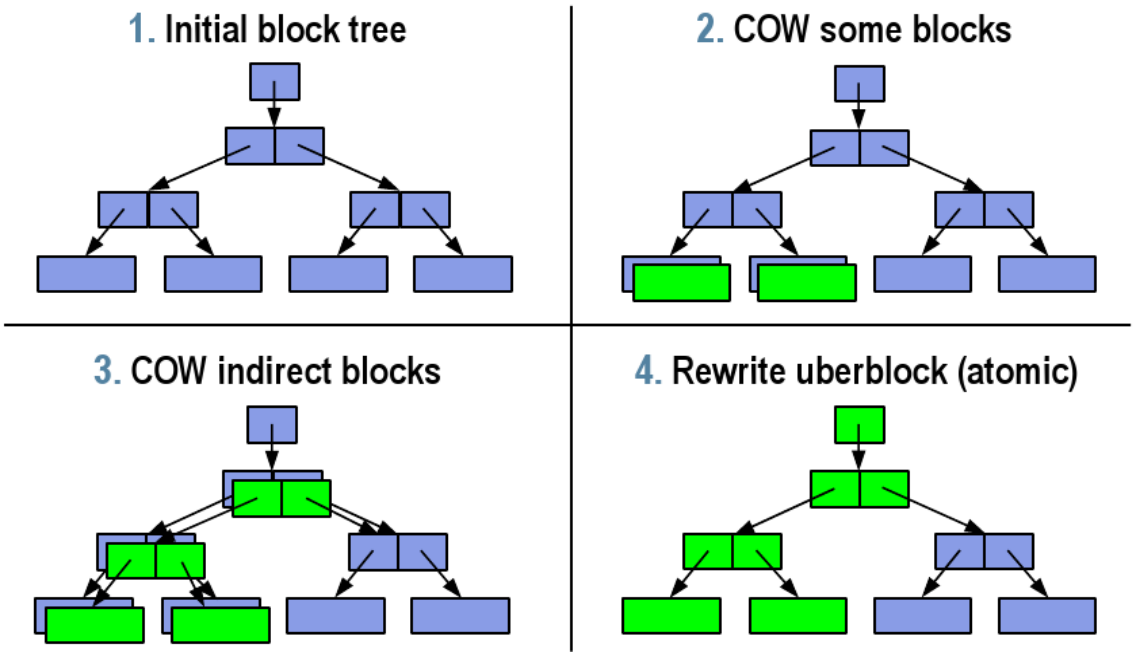
- 2^{64} — Počet *snapshot*-ov
- 2^{48} — Počet entít v jednom adresáre
- 16 EiB[1] (264 bytes) — Maximálna veľkosť partície
- 16 EiB — Maximálna veľkosť súboru
- 16 EiB — Maximálna veľkosť ktoréhokoľvek atribútu
- 256 ZiB[2] (2^{78} bajtov) — Maximálna veľkosť každého *zpool*
- 2^{56} — Maximálny počet atribútov jedného súboru
- 2^{56} — Maximálny počet súborov v jednom adresáre
- 2^{64} — Maximálny počet zariadení v jednom *zpool*
- 2^{64} — Maximálny počet *zpool* v jednom systéme
- 2^{64} — Maximálny počet súborových systémov (partícií) v jednom *zpool*

2.7.3. Transakčný model Copy-on-write

ZFS používa transakčný model s názvom copy-on-write. Všetky ukazovatele na dátové bloky (*block pointers*) v rámci súborového systému obsahujú 256-bitový kontrolný súčet príslušného bloku, ktorý je kontrolovaný pri čítaní bloku[22]. Blok obsahujúci aktívne dáta nie je nikdy priamo prepisovaný; namiesto toho je vyhradený nový blok, zmenené dáta sú zapísané do neho a všetky metadáta odkazujúce sa na blok sú jednoducho čítané, premiestňované a zapisované. Na obmedzenie preťaženia pri tomto procese sú viaceré zmeny spájané do tzv. transakčných skupín a je vytváraný tzv. *intend log* (v iných súborových systémoch tieť známy pod pojmom *žurnál*) zaznamenávajúci zamýšľané operácie a umožňujúci v prípade zlyhania nádrat k stavu pred vykonaním danej operácie.

1 1 EiB (exbibyte) je $2^{60} \approx 1.153 \cdot 10^{18}$ bajtov

2 1 ZiB (zebibyte) je $2^{70} \approx 1.181 \cdot 10^{21}$ bajtov



Obrázok 8: Fázy transakcie copy-on-write

2.7.4. Snímkovanie (snapshots) a klonovanie (clones)

Výhodou systému copy-on-write je, že po zápise nových dát, bloky obsahujúce staré dáta môžu byť zachované, čo umožňuje veľmi rýchlo (prakticky okamžite) a počas plnej prevádzky (za behu) systému vytvoriť snímku partície zaznamenávajúcu jej okamžitý stav. Tento snímkovaní systém zároveň šetrí diskovú kapacitu tým, že dáta, ktoré neboli od okamihu snímkovania zmenené neukladá dvakrát.

Môžu byť vytvorené aj prepisovateľné snímky, tzv. **klony** (*clones*), ktoré využívajú princíp vzájomného zdieľania dátových blokov medzi rôznymi partíciami[22]. Keď nastane v jednej z nich zmena, je pre danú partíciu vytvorený nový dátový blok a predošlý blok zostáva nezmenený pre ostatné partície, ktoré ho používajú.

2.7.5. Dynamické rozloženie záťaže

Dynamické rozloženie záťaže (*Dynamic striping*) maximalizuje priepustnosť systému tým, že po pridaní nového fyzického disku naň automaticky presunie časť dát a tým odľahčí ostatné fyzické disky od obsluhy operácií nad danými dátami. ZFS sa snaží dátovými presunmi medzi fyzickými zariadeniami zabezpečiť optimálne rozloženie záťaže.

2.7.6. Rýchle vytváranie súborových systémov (partícií)

V systéme ZFS manipulácie so súborovými systémami (partíciami) sú podstatne jednoduchšie ako v bežných súborových systémoch. Namiesto fyzického premiestňovania množstva dát sú do veľkej miery realizované len modifikácie príslušných odkazov. Podľa [22], čas a zložitosť vytvárania novej partície v rámci ZFS sú približne ekvivalentné vytváraniu nového adresára v niektorých bežných súborových systémoch.

2.7.7. Ďalšie vlastnosti

K ďalším zaujímavým vlastnostiam súborového systému ZFS partia:

- Komprimácia súborového systému (partície) v reálnom čase na úrovni jadra operačného systému.
- Šifrovanie[1] súborového systému (partície) v reálnom čase na úrovni jadra operačného systému.
- Možnosť uvoľňovania a pridávania fyzických diskov za behu.
- Proklamované optimálne rozloženie logických blokov v rámci fyzických zariadení eliminuje potrebu defragmentácie.
- ZFS používa variabilné veľkosti blokov až do veľkosti 128 kilobajtov.
- Viacero nezávislých tokov predvýberu (*prefetch streams*) s automatickou detekciou dĺžky.
- Paralelne realizované adresárové operácie.
- Rovnomerné rozloženie záťaže a kapacity medzi fyzickými zariadeniami.
- *Ditto* bloky - Metadáta sú replikované na viacerých miestach (ak je to možné tak na viacerých fyzických zariadeniach), takže v prípade poruchy fyzického sektora, ktorý nie je chránený redundanciou dôjde síce k strate dát, ale metadáta môžu byť stále zachránené.
- Príkaz uvolnenia vyrovnávacej pamäte (*cache flush*) vyprázdni všetky vyrovnávacie pamäte a uvedie súborový systém do konzistentného stavu, v ktorom je bezproblémovo pripravený napríklad na vypnutie systému.

1 V súčasnosti vo fáze testovania (alpha stage).

2.8. Bezpečnostné prvky OS Solaris

Základné vlastnosti v oblasti bezpečnosti operačného systému Solaris 10 sú:

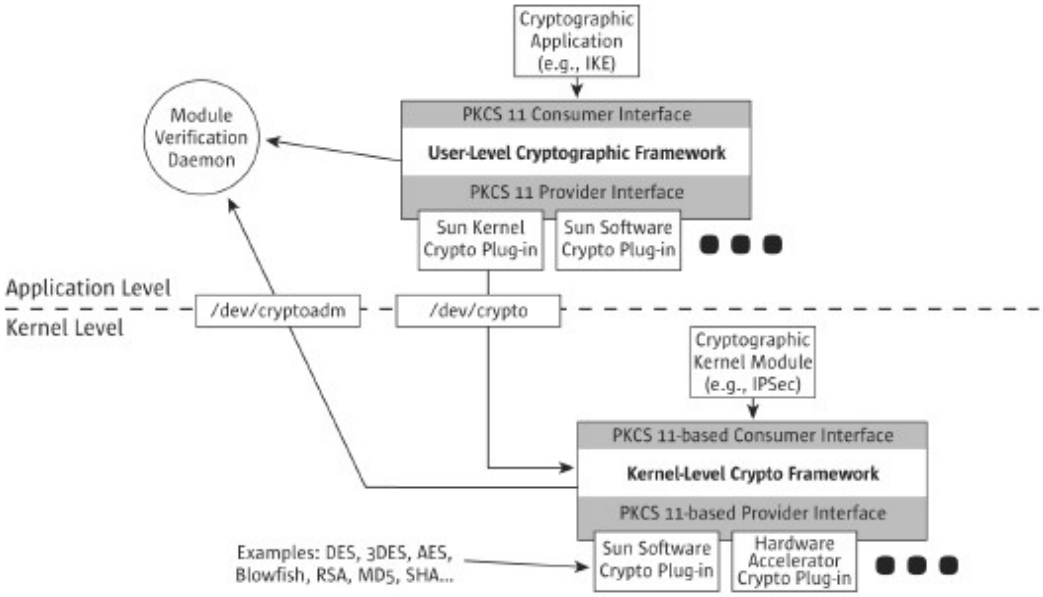
- Umožňuje silné riadenie bezpečnosti požadované najmä vládnymi a finančnými organizáciami. Toto je možné najmä vďaka bezpečnostnej nadstavbe **Solaris Trusted Extensions**.
- Overuje integritu systému s použitím **Solaris Secure Execution** a ďalších vlastností zameraných najmä na ochranu integrity súborov.
- S pomocou pokročilých možností nastavovania prístupových práv pre užívateľov a procesy je možné používať systém minimálnych privilégií.
- Zabezpečuje šifrovú ochranu dát s použitím IPSec/IKE a ochranu pred sieťovými útokmi,prípadne nastavovanie prístupových práv na úrovni siete pomocou integrovaného firewallu **Solaris IP Filter**.

2.8.1. Izolácia aplikácií pomocou Solaris zón

Zóny umožňujú veľmi dobré oddelenie aplikácií až na úrovni jadra operačného systému. Toto oddelenie je vhodné najmä z hľadiska bezpečnosti pretože v prípade, že sa útočníkovi podarí pomocou zraniteľnosti v niektorej službe/aplikácii získať nadštandardný prístup k systému, dokáže ohroziť len súčasti zóny, v ktorej sa nachádza. Viac informácií k tejto téme sa nachádza v osobitnej kapitole venovanej Solaris zónam a kontajnerom.

2.8.2. Solaris Cryptographic Framework

Solaris Cryptographic Framework poskytuje aplikáciám spoločné API pre využívanie kryptografických funkcií na úrovni systému. Toto rozhranie poskytuje jediný bod ktorý je potrebné spravovať a jednotný prístup k hardvérovým akcelerátorom šifrovania čím zvyšuje rýchlosť aplikácií využívajúcich šifrovacie mechanizmy a zabezpečuje riadenie prístupu a vyťaženia (load-balancing) daných akcelerátorov [10].



Obrázok 9: Schéma Solaris Cryptographic Framework.

2.8.3. Integrovaný firewall

V OS Solaris verzii 10 je použitý populárny firewall **IP Filter** ktorý bol vytvorený open-source komunitou. Tento firewall je plne integrovaný do častí jadra ktoré sa zabezpečujú sieťové funkcie (*Solaris IP stack*).

Tento firewall umožňuje filtrovanie prechádzajúcej sieťovej prevádzky na základe stavov spojení (statefull firewall). Umožňuje aj preklad sieťových adries (Network Address Translation) a obsahuje ochranu proti skenovaniu portov (port scan) či tzv. ICMP flood útoku. Neobsahuje však množstvo ďalších vlastností obsiahnutých napr. v rozšírenom firewalle netfilter, ktorý je bežnoý najmä v OS Linux.

2.8.4. Prednastavená bezpečnosť (security by default)

OS Solaris 10 umožňuje inštaláciu typu *The Reduced Networking Metacluster* pri ktorej je nainštalovaná len sada minimálne potrebných softvérových komponentov. K nej potom môže administrátor pridávať ďalšiu potrebnú funkcionalitu bez toho, aby boli nainštalované nepotrebné prebytočné prvky, ktoré môžu byť zneužité na bezpečnostný útok. Lokálni užívatelia systému majú plný prístup k sieťovým zdrojom, ale samotný operačný systém je zo siete takmer neviditeľný[10].

2.8.5. Bezpečnostné rozšírenie Solaris Trusted Extension

Solaris Trusted Extension je sada rozšírení štandardného OS Solaris o prídavné bezpečnostné vlastnosti. Pôvodne bola vyvíjaná ako samostatná vetva OS Solaris (vychádzajúca z OS Solaris verzie 8) určená špeciálne pre vládne inštitúcie a srístupnená aj finančným inštitúciám. Toto rozšírenie už na rozdiel od štandardného operačného systému Solaris verzie 10 nie je k dispozícii na bezplatné použitie.

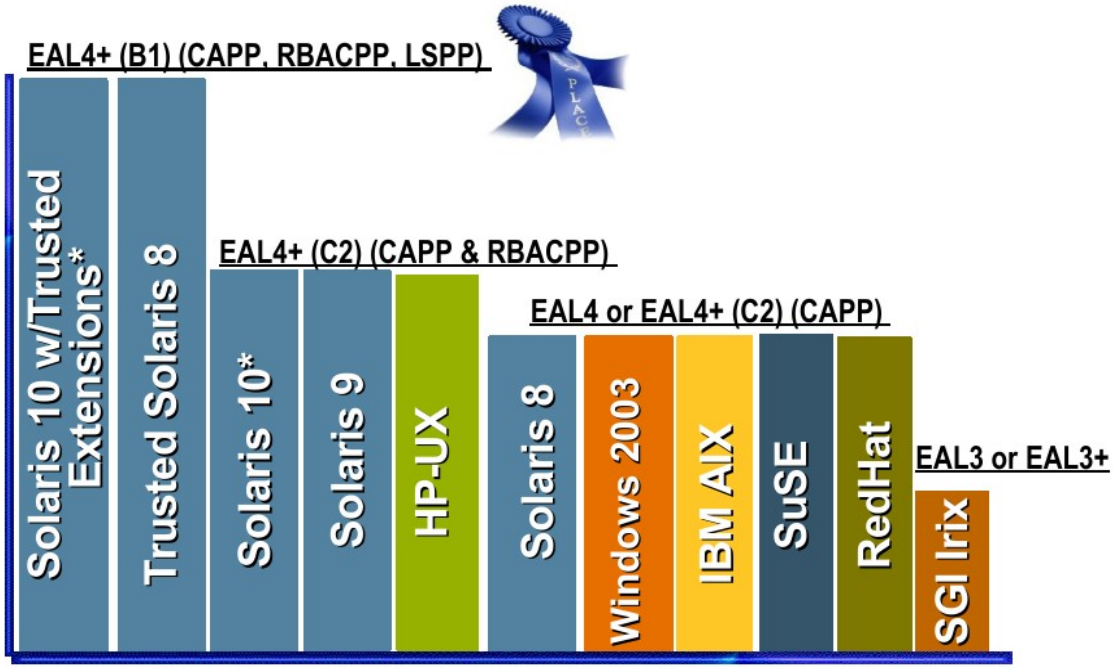
Riadi prístup užívateľov a procesov k dátam a zariadeniam pomocou špeciálneho mechanizmu riadenia prístupov [11]. Okrem spomenutého rozdeľuje štandardného superužívateľa na viacero oddelených pozícií (roles). Kombináciou označovania (labeling) systémových objektov a dát podľa stanovených základných úrovní privilégií (clearance levels) pre každého užívateľa a zaznamenávaním aktivít užívateľov výrazne zvyšuje auditovateľnosť aktivít užívateľov a možnosť spätného odkrokovania ich úkonov. Grfické používateľské rozhranie je prispôsobené na prevenciu možností odchytávania hesiel (a podobných aktivít) pomocou škodlivých kódov typu trójsky kôň (Trojan horses). Systém prístupových práv typu *Mandatory Access Control* zabezpečuje hierarchickú kategorizáciu informácií čím zvyšuje úroveň ich zabezpečena.

Operačný systém Soalris s bespečnostným rozšírením Solaris Trusted Extension je certifikovaným najvyššími bezpečnostným certifikátmi (Controlled Access Protection Profile (CAPP), Role-Based Access Control Protection Profile (RBACPP), Evaluation Assurance Level 4+ (EAL 4+)) určenými na ohodnocovanie bezpečnosti operačných systémov.

Operačný systém Trusted Solaris [11] je jediným operačným systémom ktorý bol nezávislo certifikovaný certifikátom Common Criteria Evaluation Assurance Level 4+ (EAL 4+)[1] s tromi kritickými ochrannými profilmi:

- Labeled Security Protection Profile (LSPP),
- Controlled Access Protection Profile (CAPP),
- Role-Based Access Control Protection Profile (RBACPP).

1 Táto úroveň zodpovedá bezpečnostnej úrovni známej pod označením B1. Podľa vojenskej klasifikácie informácií sa jedná o stupeň utajenia informácií typu prísne tajné.



Obrázok 10: Porovnanie bezpečnostných certifikácií operačných systémov.

* Solaris 10 3/05 a 11/06 sú v čase písania tohto dokumentu (december 2006) v štádiu certifikácie.

2.8.6. Automatizovaná správa bezpečnostných záplat

Originálne nazývaná *Automated Patch Management* [11] je systémom proaktívnej a centrálne riadenej aplikácie bezpečnostných záplat (patch) na množstvo počítačov s OS Solaris 10. Keď sa vyskytne nová zraniteľnosť, spoločnosť Sun vytvorí bezpečnostnú záplatu a aktívne informuje všetky systémy, pre ktoré je táto záplata aktuálna. Nedeje sa tak ale podľa zaužívaného modelu, keď sa operačný systém v periodických časových intervaloch sám pripája na aktualizačné centrum výrobcu a informuje sa o nových záplatách. V tomto prípade iniciatíva vychádza od výrobcu a ten automatizovaným systémom informuje servery prevádzkovateľov, ktoré sú schopné automaticky získať a nainštalovať danú záplatu. Spoločnosť Sun má informácie o nainštalovaných verziách operačného systému a prevádzkovaných aplikáciách a teda posiela len upozornenia relevantné pre konkrétnu inštaláciu. Túto službu je ale tiež potrebné dokúpiť. Štandardné bezpečnostné záplaty sú k dispozícii zdarma.

Práca s aktualizáciami a záplatami je možná prostredníctvom viacerých systémových nástrojov. Jedným z nich je *Solaris Patch Manager* ktorý je schopný automaticky stiahnuť záplaty potrebné pre daný systém. Tieto sú výrobcom poskytované vo forme of digitálne podpísaných .jar súborov. Tento systém počíta s inštaláciou záplat na mnohých prevádzkovaných serveroch podľa potreby alebo

súčasne, pričom aplikácia záplat môže byť prostredníctvom nástroja *Solaris Patch Manager* riadená z jedného počítača.

2.8.7. Prevencia útokov so zneužitím chyby pretečenia zásobníka

Útoky typu stack owerflow (prtečenie zásobníka) sú veľmi časté a dovoľujú pomerne jednoduchým a v súčastnosti už dobre zdokumentovaným spôsobom zneužiť chyby v aplikáciách na prienik do systému[11]. OS Solaris 10 poskytuje ochranu proti tomuto typu útokov pre všetky 64-bitové aplikácie a voliteľne aj pre všetky 32-bitové aplikácie. Súto ochranu je možné aktivovať jednoduchým nastavením. Základné administračné nástroje majú túto ochranu aktivovanú automaticky. Vývojári aplikácií môžu takto zabezpečené knižnice OS Solaris prilinkovať do ich aplikácií čím zvýšia ich ochranu proti spomenutým zraniteľnostiam. Táto funkcionalita je dostupná pre procesory SPARC aj AMD64.

2.8.8. Integrita súborov a bezpečné spúšťanie procesov

Spoločnosť Sun digitálne podpisuje skoro všetky produkované binárne spustiteľné súbory [5] systému Solaris 10. Toto umožňuje administrátorovi sledovať zmeny daných binárnych súborov, tieto zmeny často indikujú realizovaný bezpečnostný útok. V budúcich verziách OS Solaris je plánovaná možnosť zamknutia systému spôsobom umožňujúcim spúšťanie len správne[1] digitálne podpísaných binárnych súborov.

V OS Solaris sa tiež vyskytuje nová aplikácia zabezpečujúca kontrolu integrity dátových súborov a používateľských aplikácií známa pod označením *Basic Audit and Reporting Tool* (BART). Sun tiež poskytuje kontrolné súčty (hashes) všetkých súborov dodávaných spolu so systémom Solaris 10 ako súčasť projektu *Solaris Fingerprint Database*.

1 Operačný systém bude automaticky kontrolovať digitálne podpisy oproti zoznamu dôveryhodných podpisových autorít.

3. Posudzovanie operačných systémov

Kapitola analyzuje rôzne možnosti a aspekty posudzovania kvality a vhodnosti operačného systému, ako základného softvérového komponentu. Metódy posudzovania kvality operačných systémov sú potrebné pre výber vhodného operačného systému pre konkrétnu aplikáciu vzhľadom na požiadavky budúceho prevádzkovateľa.

Pri posudzovaní sa bude klásť dôraz na OS Solaris, kde budú uvádzané príklady realizovaných porovnávaní.

3.1. Riadiaci versus technickí pracovníci

Tento výber býva často realizovaný najmä manažérmi, prípadne inými vyššie postavenými osobami, ktorí vykonávajú zásadné rozhodnutia no neprichádzajú do priameho styku s konkrétnymi technickými problémami a úlohami. Toto je najmä vo veľkých organizáciách pomerne častá chyba, pretože toto rozhodnutie často naráža na odpor technického personálu. Je preto potrebné, aby sa budúci priami používatelia priamo zúčastňovali na výbere softvérového vybavenia, s ktorým budú pracovať (a ktorého najdôležitejším základom spolu s hardvérom je operačný systém) a z ich hľadiska posúdili vyberaný produkt. Zároveň je potrebné, aby sa pri výbere stretli požiadavky manažmentu, ktorý rozhoduje napríklad o financiách a udržiava filozofiu spoločnosti spolu s požiadavkami technických pracovníkov, ktorí technicky zabezpečujú chod kritických obchodných služieb.

3.2. Metódy porovnávania aspektov kvality operačných systémov

V prípade reálneho rozhodovania, posudzovania alebo vyberania existujúceho produktu (v tomto prípade operačného systému) existuje niekoľko základných metód, ktoré môžu byť využité na ohodnotenie základných vlastností. Najväčším spoločným problémom všetkých metód je ich objektivita, prípadne relevantnosť získaných výsledkov. Všetky metódy (vrátane porovnávacích meraní (benchmark))

sú totiž silne závislé od zamerania, prípadne zvoleného konkrétneho spôsobu merania. Praktické skúsenosti sa často zásadne líšia od nameraných hodnôt a pre reálneho prevádzkovateľa sú dôležité práve spomenuté praktické výsledky.

Základné spôsoby posudzovania vlastností produktov sú:

- porovnávacie merania – benchmarks,
- porovnávanie zverejnených informácii,
- vlastné testovanie,
- názory iných používateľov.

3.2.1. Porovnávacie merania

Sú používané najmä pri meraní rýchlostí. Prebiehajú tak, že sa na meraný operačný systém nainštaluje program, ktorý mnohonásobne s rôznymi variáciami vykonáva zvolené operácie a meria čas (prípadne iné parametre) ich vykonávania. V prípade sieťových testov sa jedná napríklad o reakcie na rýchlosť odozvy, prípadne na množstvo spracovaných sieťových požiadaviek za daný časový interval. Pri sieťových meraniach spravidla nie je potrebné na meraný systém inštalovať žiadny špeciálny softvér.

Pri týchto testoch je dôležitý spôsob zaťaženia meraného systému. Z tohto hľadiska ich môžeme rozdeliť na:

1. benchmark testy základných operácii OS
2. testy simulovanej prevádzky

Testy základných operácií (napríklad otváranie a zatváranie súborov, sieťových socketov, pamäťové operácie, vytváranie procesov a pod.) sú najjednoduchšie no zároveň najvzdialenejšie od požiadaviek reálnej prevádzky.

Pri **testoch simulovanej prevádzky** už pracujeme s konkrétnymi službami (napr. web server v kombinácii s databázovým serverom), ktoré zaťažujeme množstvom požiadaviek podobných reálnej prevádzke. Tento test môže ohodnotiť daný systém pomerne neobjektívne, no naopak je vhodný na ohodnotenie vhodnosti systému na konkrétny účel použitia.

Výhody

- výstupom sú presné hodnoty

- je možné jednoducho porovnať konkurenčné výsledky
- existujú hotové softvérové produkty, s ktorými je možné relatívne jednoducho tieto merania vykonávať

Nevýhody

- pri zložitých systémoch sa výsledky zásadne menia vzhľadom od iných parametrov
- pri zložitých systémoch nemusia výsledky reprezentovať reálnu prevádzku
- rýchlosť nemusí byť tým najdôležitejším parametrom

Existujúce aplikácie vykonávajúce porovnávacie (benchmark) testy

Existuje mnoho aplikácií realizujúcich výkonové testy počítačov. Väčšinou sa však zameriavajú na meranie výkonu hardvéru a nie je jasná ich spojitosť v meraním výkonu operačného systému. Vychádzajú z predpokladu, že daný program bude pod tým istým operačným systémom spustený na viacerých rôznych hardvérových konfiguráciách a výsledky merania ukážu výkonnostné rozdiely medzi nimi. Meranie výkonu operačných systémov však vychádza z opačného predpokladu, ktorým je inštalácia rôznych operačných systémov na rovnakom hardvéri a následné porovnanie výkonnosti. Ich ďalšou zásadnou nevýhodou je, že väčšinou nebývajú vyvíjané pre rôzne operačné systémy, čo samo o sebe vylučuje ich použitie na porovnávanie výkonnosti operačných systémov.

Program sysbench

Tento program je príkladom nástroja určeného na porovnávacie merania výkonnosti operačných systémov. Pomocou neho bolo realizované napríklad aj meranie výkonu databázových aplikácií uvádzané v tejto diplomovej práci... Jeho hlavnou výhodou je pomerne široké spektrum deklarovaných testovaných funkcií a viacvláknový beh. Jeho nevýhodami naopak sú silné naviazanie na databázové aplikácie a stagnácia jeho vývoja (posledná aktualizácia uvádzaná na stránke projektu je z 7. októbra 2004).

3.2.2. Porovnávanie zverejnených informácii

Základným a väčšinou prvým spôsobom porovnávania vyberaných produktov, ktorý vykonávame v podstate automaticky je porovnanie konkurenčných produktov na základe informácií, ktoré o nich máme. Väčšinou sa porovnávajú zverejnené vlastnosti a podobné parametre

Základné parametre, ktoré je možné porovnávať sú **cena**, **funkcie**, vlastnosti, podpora od dodávateľa a pod. Bezpečnosť je možné porovnávať napríklad **množstvom** a závažnosťou nájdených **bezpečnostných nedostatkov** a rýchlosťou ich odstránenia výrobcom.

Špeciálnym prípadom sú **certifikácie**. Nezávislé organizácie zverejnia presné kritéria a na ich základe realizujú dôkladné testovanie produktov. Ak produkt splní požiadavky, býva mu vystavený certifikát, ktorý dáva spotrebiteľovi záruku istej úrovne produktu v závislosti od spôsobu a náročnosti merania. Z hľadiska operačných systémov sú dôležité najmä bezpečnostné certifikáty.

Kombináciou tohto a predchádzajúceho spôsobu posudzovania kvality sú existujúce porovnávacie merania, ktoré bývajú realizované priamo výrobcami daných produktov, prípadne nezávislými organizáciami. Výrobca má prirodzenú snahu o to, aby jeho produkt vyšiel z merania s najlepšími výsledkami a preto často prispôsobuje merania, alebo dokonca modifikuje výsledky podľa svojich potrieb. V prípade tretích strán aj naďalej zostáva otázna ich nezávislosť, pretože sa zvyčajne nejedná o veľké organizácie, ktoré je ťažké ovplyvniť, ale skôr o menšie spoločnosti realizujúce merania na objednávku (napríklad redakcie technických časopisov).

Výhody

- jednoduchý prístup k výsledkom
- minimálna prácnosť a časová náročnosť
- merania špecializovaných organizácií bývajú dôsledné a podrobné
- merania sú často realizované skúsenými odborníkmi

Nevýhody

- je obtiažne zistiť objektívnosť zverejňovaných informácií
- výrobca prispôsobuje zverejňované informácie svojim potrebám
- výrobca často neudáva dostatočne podrobné a relevantné technické informácie

3.2.3. Vlastné testovanie

Je najspoľahlivejším, no zároveň najprácnejším spôsobom posúdenia vhodnosti posudzovaného produktu. Testovanie realizované budúcim potenciálnym používateľom mu najlepšie ukáže vlastnosti, ktoré sú pre neho kľúčové a na základe ktorých sa rozhoduje. V prípade vlastných testov nehrozí ani manipulácia výsledkov. Tieto testy ale zaberajú množstvo času a aj napriek tomu väčšinou nedokážu otestovať produkt z dlhodobého hľadiska. Na takéto testovanie je potrebné vyhradiť technické, personálne aj časové zdroje. Hrozí aj ovplyvnenie testera tým, že keď sa mu po istej investovanej námahe a čase podarí spojazdniť testovaný produkt do požadovaného stavu, je spokojný a necíti potrebu namáhavo testovať aj ďalšie produkty, ktoré by ale mohli vykazovať lepšie výsledky. V opačnom prípade zas tester subjektívne porovnáva množstvo investovaného času a námahy do sfunkčnenia produktu, namiesto vlastností jeho dlhodobej prevádzky.

Výhody

- je možné otestovať produkt priamo z hľadiska potrieb používateľa
- už počas testovania je možné naučiť sa pracovať s produktom, prípadne ho pripraviť na reálne používanie v prostredí, do ktorého má byť nasadený
- výsledky testovania nebudú manipulované
- vlastné skúsenosti s testovaným produktom sú najobjektívnejším ukazovateľom
- je vhodné na posudzovanie jednoduchosti a vlastnosti užívateľských rozhraní

Nevýhody

- časová aj personálna náročnosť
- nie je možné posudzovať vlastnosti z dlhodobého hľadiska
- je obtiažne otestovať stabilitu produktu
- je obtiažne zaistiť kvalitu testovania

3.2.4. Názory používateľov

Vďaka internetovým diskusným fóram je možné získať priame a nezmanipulované názory osôb, ktoré prišli do osobného styku s porovnávaným produktom, prípadne ho sami dlhodobo používali. Zároveň ale vyvstáva otázka ich spôsobilosti,

objektívnosti a dôveryhodnosti. V rôznych používateľských komunitách je často udržiavaný jednotný názor väčšiny, možno teda povedať, že členovia spomenutých skupín sú ovplyvňovaní. Títo používatelia môžu byť aj neskúsení, prípadne môžu podávať nepresné informácie. V každom prípade sa však jedná o veľmi subjektívny názor, ktorý ale narozdiel od predchádzajúceho popisovaného spôsobu nie je výhodný pre cieľovú organizáciu (iní používatelia posudzujú produkt z hľadiska ich potrieb, nie potrieb vyberajúcej organizácie). Pri skúmaní väčšieho množstva skupín (vrátane komunít združených okolo konkurenčných systémov) a ohodnotení závažnosti jednotlivých názorov (napr. braní do úvahy len názorov ľudí, ktorí so systémom priamo dlhodobejšie pracovali) je možné získať dobrú reprezentatívnu vzorku. Pri výbere operačného systému často krát rozhoduje práve odporúčanie profesionálov, ktorí majú so systémom (a konkurenčnými systémami) dlhodobé priame a osobné skúsenosti.

Výhody

- je možné získať informácie o dlhodobej prevádzke
- skúsení používatelia dokážu upozorniť na skryté problémy
- používatelia uvádzajú skúsenosti z reálnej praxe
- informácie nie sú manipulované výrobcom

Nevýhody

- veľká subjektivita
- otázna spoľahlivosť a dôveryhodnosť informácií
- nie je možné použiť pri úplne nových produktoch
- informácie zväčša nebývajú relevantné pre vyberajúcu organizáciu
- popri zaujímavých informáciách sa vo fórach vyskytuje množstvo irelevantných vyjadrení

3.3. Aspekty kvality operačných systémov

Pri výbere operačného systému pre konkrétnu aplikáciu (účel použitia) je potrebné stanoviť požiadavky danej aplikácie a ich priority z pohľadu prevádzkovateľa. V tejto kapitole budeme vychádzať zo zoznamu najdôležitejších vlastností operačných systémov, ktorý je vysvetlený v kapitole Serverové operačné systémy. Spomínané najdôležitejšie vlastnosti sú:

- jednoduchosť správy
- bezpečnosť
- stabilita
- vlastnosti (features)
- výkon
- hardvérové požiadavky, hardvérová kompatibilita
- rozšíriteľnosť
- náklady

Jednoduchosť správy

Na prvý pohľad by bolo možné porovnávať priateľskosť používateľského rozhrania, intuitívnosť, či vizuálnu príťažlivosť grafického používateľského rozhrania (**GUI**) no práve skúsení odborníci uprednostňujú **rozhranie príkazového riadku**. Tieto sa medzi sebou tiež líšia, no rozdiely sú pomerne malé a možnosti ich prispôsobenia (najmä v prípade UNIXových operačných systémov) zas veľmi široké. Preto je potrebné jednoduchosť správy porovnávať najmä zo subjektívneho hľadiska budúceho používateľa a to z ohľadom na spôsob práce, na ktorý je daný používateľ zvyknutý a na to, s akým systémom používateľ už vie pracovať. Veľmi často totiž nad kvalitou víťazí práve **návyk používateľa** a fakt, že s daným systémom vie plynulo pracovať, zatiaľ čo s iným by sa musel učiť.

Z tohto hľadiska je pre OS Solaris od verzie 10 pozitívne, že sa v maximálnej možnej miere snaží priblížiť už pomerne rozšírenému (najmä v cieľovej skupine používateľov OS Solaris) operačnému systému Linux.

Bezpečnosť

V prípade rozšírených operačných systémov je ťažké kvantifikovať úroveň ich bezpečnosti. Veľmi závisí od ich aktuálneho nastavenia a správnosti ich používania či od vlastností ich správcov, používateľov alebo útočníkov. Rozšírený operačný systém je vystavený oveľa väčšiemu množstvu útokov no zároveň to zvyšuje jeho bezpečnosť. Operačný systém nasadzovaný na najdôležitejšie aplikácie býva zas často chránený dodatočnými bezpečnostnými opatreniami, ale je zároveň príťažlivejší pre útočníkov, ktorí sa k nemu ale ťažšie dostanú dostatočne blízko. Veľmi diskutabilná je tu otázka otvorenia, či utajenia zdrojových kódov daného systému. V otvorených zdrojových kódoch je možné oveľa ľahšie objaviť

bezpečnostné slabiny a nahlásiť ich výrobcovi, ktorý ich môže opraviť. No vo väčšine prípadov spomenuté slabiny najprv objavia útočníci, ktorí ich na niekoľko prípadov zneužijú a až potom dôjde k ich oprave.

Jednou z možností merania bezpečnosti operačných systémov je spočítanie **množstva známych zraniteľností** za určité časové obdobie. Pre rôzdne operačné systémy takéto prieskumy bývajú zverejňované napríklad spoločnosťou Symantec v pravidelných správach nazvaných *Internet security threat report*. Podobné správy generuje spoločnosť IBM (prostredníctvom akvizovanej ISS) kde sa nazývajú *IBM Internet Security Systems X-Force Trend Statistics*.

Bezpečnosť je možné ďalej posúdiť aj pomocou **penetračného testovania**, kde je možné zisťovať napríklad odolnosť operačného systému proti sieťovým detektorom, skenom, alebo útokom zamietnutia služby (DOS útoky).

Stabilita

Stabilitu operačného systému je možné kvantifikovať podobným spôsobom, ako stabilitu iných technických zariadení a teda parametrom **strednej doby medzi poruchami**. Zas je ale obtiažne rozlíšiť medzi udalosťami, ktoré je ešte možné považovať za poruchy a tými, ktoré nie. Poruchy sú silne závislé aj od použitého hardvéru či aktuálnej prevádzke a vyťažení počítača. V prípade správneho používania je pravdepodobnosť poruchy samozrejme nižšia, ako v prípade nesprávneho používania, no v praxi je prakticky nereálne zabezpečiť 100% správnosť používania a teda rozhoduje rozsah intervalu, v ktorom sa môže používateľ pohybovať (laicky povedané: čo všetko si ešte môže dovoliť) aby systém pracoval správne.

Na posudzovanie tohto kritéria je (najmä z dlhodobého hľadiska) najvhodnejšie vychádzať z reálnych skúseností mnohých používateľov medzi ktorými sa štatisticky stratia rozdiely v hardvéri a spôsobe používania. V prípade operačných systémov však aj tento parameter zostáva pomerne subjektívnym.

Vlastnosti (features)

Pri tomto parametre tržba rozlišovať medzi tým, čo patrí medzi vlastnosti a funkcionalitu operačného systému a tým, čo je pokryté ďalšími softvérovými komponentami, ktoré sú dodávané spolu s operačným systémom, prípadne je o ne daný OS rozšíriteľný. Dôležité sú aj podmienky (napríklad cenové) rozšíriteľnosti daného OS o spomenuté komponenty. Operačné systémy UNIXového typu bývajú

spravidla distribuované spolu s mohutnou zásobou softvérových komponentov ktorými je možné pokryť prakticky každú oblasť použitia. Operačný systém Windows býva naopak distribuovaný len so základnou množinou nástrojov s veľmi obmedzenou funkcionalitou. Spomenutá vlastnosť zároveň ovplyvňuje aj jednoduchosť správy systému.

Najlepším spôsobom porovnania tohto kvalitatívneho aspektu OS je jednoduché vymenovanie spolu s kvantifikáciami úrovní menovaných funkcií a podmienkami možností ich využívania v danom prostredí.

Výkon

Pod výkonom sa najčastejšie rozumie rýchlosť, prípadne množstvo vykonaných požadovaných operácií za daný čas. Tento je možné (za daných podmienok) pomerne jednoducho a presne kvantifikovať benchmark meraniami. V praxi však (najmä s poklesom cien hardvéru) požiadavky na rýchlosť operačného systému strácajú na dôležitosti a používatelia sa stále viac zameriavajú na iné spomenuté vlastnosti. Viac informácií o meraniach rýchlosti a súvisiacich aspektoch sa nachádza v predchádzajúcej kapitole v časti *Porovnávacie merania*.

Hardvérové požiadavky, hardvérová kompatibilita

Túto je možné porovnať pomerne presne, pretože výrobca daného systému väčšinou jasne udáva informácie o požidavkach na hardvér. Často sú udávané minimálne aj odporúčané hardvérové požiadavky.

Rozšíriteľnosť

Tento aspekt priamo súvisí s vlastnosťami (features) operačného systému. Pre ďalšie informácie viz. časť vlastnosti v tejto kapitole.

Náklady

Podobne ako pre hardvérové požiadavky, aj pre túto vlastnosť operačného systému platí, že je pomerne jednoducho zistiteľná aj porovnateľná s konkurenciou. Okrem samotnej ceny za operačný systém je potrebné brať do úvahy aj ceny za:

- záruku
- podporu od dodávateľa

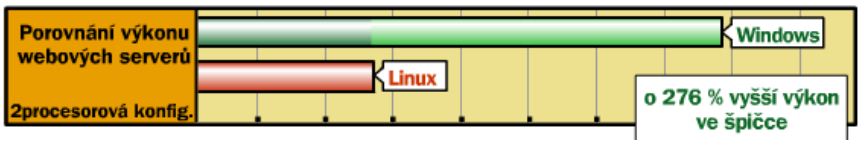
- dokumentáciu
- prídavné komponenty
- potrebný hardvér
- licencie na konkrétny spôsob nasadenia
- inštaláciu a správu
- iné súvisiace náklady

V prípade prechodu na iný operačný systém je potrebné zohľadňovať aj náklady na samotný prechod pozostávajúce napr. z:

- inštaláciu nového systému
- inštaláciu a sfunkčnenie služieb
- migráciu dát
- testovanie a stabilizáciu systému
- vzdelávanie zamestnancov
- straty z výpadku služieb
- ďalšie náklady

3.4. Diskutabilnosť výsledkov merania

Výsledky kvantitatívneho merania vybraných parametrov operačných systémov bývajú často diskutované a spochybňované. Príkladom môže byť kampaň *Get the facts* organizovaná spoločnosťou Veritest a sponzorovaná spoločnosťou Microsoft. Táto štúdia porovnávala výkon webového servera Microsoft IIS bežiacom na operačnom systéme Windows 2003 Server a webového servera Apache bežiacom na systéme Red Hat Linux. Z výsledku merania vyplynulo, že konfigurácia s operačným systémom Microsoftu mala o 276 % vyššiu maximálnu hodnotu priepustnosti než Apache [24].



Obrázok 11: Propagačný výsledok porovnávania výkonu webových serverov na báze OS Windows a Linux.

Podrobný popis merania sa nachádza v [25].

Táto analýza vzbudila veľké podozrenie a vlnu odporu v radoch zástancov operačného systému Linux a najmä kritiku objektivity testu a nastavení testovaných systémov. Analýzu a zdôvodnenie neobjektivity kampane je možné nájsť v [24].

Bezpečnosť OS Solaris

Po hlbšej analýze sa ukázalo hodnotenie množstva zraniteľností a rýchlosť uvoľňovania ich opráv ako pomerne objektívny spôsob hodnotenia bezpečnosti operačných systémov. Podľa výsledkov takýchto porovnaní, ktoré sú uvedené v tomto dokumente sa operačný systém Solaris javí ako najmenej bezpečný. Všeobecná mienka o ňom naopak hovorí ako o najbezpečnejšom a často býva nasadzovaný práve na tie najdôležitejšie (tzv. Mission Critical) servery spoločností.

4. Návrh

Kapitola rozoberá návrh postupu prác ktoré by mali byť realizované v rámci tohto projektu. Tieto vyplývajú z cieľov projektu stanovených v úvode dokumentu. Špeciálne sa venuje sumarizácii požiadaviek na vypracovávaný softvérový nástroj.

4.1. Výstupy projektu

Výstupmi projektu by mali byť:

- materiál popisujúci OS Solaris ako celok a jeho vybrané vlastnosti
- všeobecná metodika hodnotenia a výberu operačného systému
- porovnanie operačného systému Solaris s konkurenčných operačných systémov
- sumarizácia existujúcich testov OS Solaris
- implementácia nástroja na kvantitatívne hodnotenie výkonu operačného systému
- používateľská príručka jednoducho popisujúca základné a vybrané pokročilé konfigurácie operačného systému Solaris
- štatistický prieskum názorov a preferencií systémových administrátorov
- nástroje kompenzujúce vybrané nájdené nedostatky OS Solaris

4.2. Použité metódy

Na posudzovanie vybraných vlastností operačného systému Solaris ako aj systému ako celku budú použité najmä metódy:

- Štúdium oficiálnej dokumentácie výrobcu
- Vlastné testy a experimenty
- Nezávislé merania
- Vlastné skúsenosti
- Analýza ohlasov používateľskej komunity

4.3. Postup prác na projekte

Počas vypracovávania projektu (v rámci všetkých troch semestrov) by mali byť na projekte vykonané nasledovné práce:

1. Teoretická príprava.
    a. preskúmanie operačného systému ako celku, jeho vlastností a možných aplikácií v reálnej praxi
2. Inštalácia a reálna prevádzka.
    a. inštalácia operačného systému na reálnom serveri
    b. inštalácia a sprístupnenie základných aj pokročilých sieťových služieb na danom serveri a sprístupnenie daných služieb reálnym používateľom
    c. a reálna prevádzka servera v prostredí počítačovej siete
    d. praktické preskúmanie operačného systému ako celku, jeho vlastností
3. Výber a podrobnejšia analýza jeho zaujímavých pokročilých vlastností.
4. Sumarizácia a spracovanie získaných informácií (vytvorenie kapitoly Analýza).
5. Vypracovanie metodiky hodnotenia operačných systémov.
6. Zozbieranie a sumarizácia existujúcich testov a hodnotení skúmaného operačného systému.
7. Realizácia vlastných testov a implementácia príslušného softvérového nástroja na tento účel.
8. Implementácia ďalších vybraných softvérových nástrojov.
9. Na základe osobných skúseností a odskúšaných postupov vypracovanie príručky popisujúcej základné a vybrané pokročilé konfigurácie operačného systému Solaris.
10. Vypracovanie príslušnej dokumentácie.
11. Publikovanie získaných vedomostí.

Podrobnejšie informácie o spomenutých fázach projektu sú uvedené v príslušných kapitolách.

4.4. Nástroj na porovnávacie meranie výkonu operačného systému

Navrhujem vytvoriť aplikáciu demonštrujúcu možnosti kvantifikácie vybraných vlastností operačných systémov. Aplikácia by sa mala zameriavať na meranie rýchlostí základných operácií operačného systému.

4.4.1. Požiadavky

Na navrhnutý nástroj sú kladené tieto požiadavky:

- meranie rýchlosti základných operácií operačného systému
- možnosť portovania na čo najviac operačných systémov
- jednoduché, otvorené a prehľadné zdrojové kódy umožňujúce pridávanie ďalších funkcií, prípadne modifikáciu súčasných
- čo najužšia naviazanosť priamo na operačný systém (pomerne nízko úrovňový programovací jazyk)
- štatistické vyhodnotenie výsledkov mnohých meraní (rýchlosti jednotlivých operácií sa vypočítajú ako priemer z väčšieho počtu meraní)
- merania realizované so širokou škálou parametrov jednotlivých funkcií (dátové operácia realizované nad blokmi dát rôznych veľkostí)

5. Implementácia

Kapitola popisuje niektoré produkty vypracované v rámci tejto diplomovej práce.

5.1. benchmark - nástroj na porovnávacie meranie výkonu operačného systému

Aplikácia benchmark je implementovaná ako ukážka možnosti kvantifikácie (merania) parametrov operačného systému. Je vytvorená tak, aby bola jednoducho implementovateľná na mnohých operačných systémoch a tým umožňovala ich porovnávacie merania. Merania sú založené na štatistickom vyhodnocovaní rýchlosti obsluhy vybraných základných operácií operačného systému.

Ukážka meraní realizovaných nástrojom *benchmark* sa nachádza v kapitole *Hodnotenia operačného systému Solaris*, časti *Vlastné merania rýchlosti*.

5.1.1. Princíp činnosti

Program realizuje zadaný počet operácií s náhodnými veľkosťami parametrov. Napríklad realizuje zápis do zadaného množstva súborov, no do každého zapíše náhodné množstvo dát. Tieto množstvá sa v rámci zadaného rozsahu zásadne líšia, čo umožňuje otestovať systém na celých rozsahoch. Každá operácie je realizovaná ako funkcia programovacieho jazyka a je meraná dĺžka jej vykonávania. Operácie sú navrhnuté tak, aby boli navzájom nezávislé, čiže napríklad otvorenie súboru sa testuje vždy spoju s jeho zatvorením.

Základné funkcie operačného systému

Pri výbere základných funkcií operačného systému sa vychádzalo zo základných úloh operačného systému. Z hlavných oblastí, ktoré operačný systém obsluhuje boli vybraté:

- správa pamäte
- správa procesov
- súborový systém

Ako základné operácie operačného systému boli zvolené:

- správa pamäte
    - alokácia pamäte
    - realokácia pamäte
    - čítanie z pamäte
    - zápis do pamäte
- správa procesov
    - vytváranie procesov volaním služby fork
    - spúšťanie procesov príkazom system
- súborový systém
    - vytváranie súborov
    - prepisovanie súborov
    - dopĺňanie obsahu na koniec súborov
    - čítanie zo súborov
    - mazanie súborov

Režimy činnosti

Aplikácia je schopná pracovať v dvoch režimoch činnosti. Tieto sú popísané v nasledovných častiach.

Deterministický režim

V tomto režime činnosti sú merané množstvá rovnakých operácií vykonávaných za sebou. Tento režim činnosti otestuje všetky testované operácie, no v presne zadefinovanom poradí. Meranie vykonávania vybranej operácie sa začne až po dokončení všetkých meraní predošlej operácie. Program napríklad otestuje najprv vytvorenie N súborov, potom zápis do všetkých z nich, čítanie a nakoniec vymazanie všetkých z nich.

Stochastický (náhodný) režim

Tu sú vykonávané operácie vyberané náhodne, čiže oblasti vyťaženia a poradie merania sa náhodne striedajú. Nedochádza teda k sústavnému zaťaženiu jednou operáciou. Tu program napríklad vytvorí jeden súbor, alokuje pamäť, prečíta súbor, zmaže súbor, vytvorí dva procesy, vytvorí ďalší súbor, realokuje pamäť a pod.

Časy vykonávania jednotlivých operácií sa merajú osobitne, čiže okolité operácie by nemali mať vplyv na meranie času vykonávania danej operácie. Z titulu operačných systémov je ale možné, že budú mať vplyv na rýchlosť vykonávania danej operácie.

5.1.2. Kompatibilita

Keďže rýchlosť operácií je mnohokrát priamo úmerná veľkostiam dátových blokov, s ktorými sa pri meraní pracuje a tieto sú určované náhodným generátorom, je dôležité, aby boli pri všetkých meraniach generované rovnaké pseudonáhodné postupnosti čísel. Kvalita náhodnosti tu nie je dôležitá, no je nutné, aby bol generátor pseudonáhodných čísel inicializovaný vždy tou istou hodnotou. V tomto prípade je generátor napevno inicializovaný číslom 0.

5.1.3. Popis zdrojových kódov

Program pozostáva zo súborov:

- benchmark.c
- benchmark.h
- linux.c
- linux.h
- windows.c
- windows.h
- dummy.c

Súbory **benchmark.c** a **benchmark.h** obsahujú implementáciu samotných meracích algoritmov, používateľského rozhrania a prezentáciu výsledkov meraní.

Súbory **linux.c**, **linux.h**, **windows.c**, **windows.h** obsahujú implementáciu jednotlivých elementárnych funkcií priamo pre dané operačné systémy.

Súbor **dummy.c** slúži na vytvorenie prázdneho procesu, ktorý je používaný na testovanie rýchlosti spúšťania procesov.

Zdrojové kódy sú detailnejšie popísané priamo ich komentármi.

5.1.4. Možnosti rozšírenia

Aplikácia má viacero možností vylepšenia. Okrem všeobecného vylepšenia používateľského rozhrania (pridanie viacerých možností nastavenia, podrobnejšie analýzy a výstupy do ďalej spracovateľných formátov) je možné zamerať sa aj na meranie doteraz nepokrytých oblastí ako napríklad:

- meranie efektivity prepínania procesov,
- meranie obsluhy sieťových spojení,
- viacvláknový beh (prípadne súbežný beh viacerých meracích programov).

5.2. Používateľské návody

Počas testovania operačného systému Solaris bolo v rámci tejto diplomovej práce vytvorených niekoľko jednoduchých používateľských návodov, ktoré boli prostredníctvom systému wiki uverejnené ako verejne dostupné webové stránky. Zjednodušene vysvetľujú vykonávanie niektorých operácií špecifických pre OS Solaris a ukazujú rýchle spôsoby, ako tieto operácie realizovať. Návody sú písané voľným štýlom (neoficiálnou rečou) tak, aby boli čo najjednoduchšie pochopiteľné. Niektoré sú písané v Slovenskom a niektoré v Anglickom jazyku, najmä v závislosti od obdobia, v ktorom boli vytvorené.

Konkrétne boli vytvorené návody na:

1. konfiguráciu a správu zón
2. konfiguráciu a správu súborového systému ZFS
3. aktiváciu a nastavenie sieťového filtra (firewall) a prekladu sieťových adries (NAT)
4. inštaláciu a konfiguráciu systému opensource balíčkov
5. aktualizáciu a distribúciu záplat (patch)
6. ďalšie tipy a triky
7. poznámky k správe služieb a SUN špecifickému hardvéru
8. post-inštalačnú a konfiguračnú procedúru

Plné verzie týchto návodov sa nachádzajú v príslušných prílohách dokumentu.

5.3. Post-inštalačný skript

Počas testovania operačného systému Solaris bol v rámci tejto diplomovej práce vytvorený aj skript realizujúci iniciálne nastavenia potrebné po inštalácii nového systému a významnou mierou zvyšujúce používateľský komfort a jednoduchosť správy systému. Bol navrhnutý na základe vybraných nedostatkov nájdených pri inštalácii OS Solaris na reálne servery.

Keďže skript vykazoval značnú chybovosť v závislosti od rozdielnych prvých úkonov a potrieb jednotlivých serverov, odporúča sa radšej postupovať podľa návodu zverejneného na adrese http://jeronimo.ynet.sk/howto/solaris/postinstall. Tento podrobnejšie vysvetľuje úkony realizujúce skriptom a umožňuje správcom lepšie si uvedomiť ich kroky. Upravený skript je ale naďalej k dispozícii a možné ním významne urýchliť post-inštalačnú procedúru.

Presný popis činnosti skriptu a zároveň návod na jeho použitie skriptu sa nachádza v prílohe dokumentu.

5.3.1. Skript ošetrujúci závislosti originálnych balíčkov

Spolu s predchádzajúcim skriptom bol vytvorený aj ďalší ošetrujúci závislosti originálnych balíčkov výrobcu[1]. Výrobca OS Solaris v súčasnosti neposkytuje nástroj so spomenutou funkcionalitou, čo zásadne sťažuje dodatočne inštalácie, prípadne odstraňovanie väčšiny komplexnejších softvérových komponentov. Dôsledné odstránenie balíčka dokonca neumožňuje ani nástroj vytvorený opensource komunitou na správu balíčkov tretích strán.

Funkciami skriptu sú:

- vytvorenie databázy závislostí nainštalovaných balíčkov
- výpis balíčkov, ktoré sú závislé na zadanom balíčku
- výpis balíčkov, na ktorých je závislý zadaný balíček
- inštalácia zadaného balíčka s ošetrením závislostí (inštalácie balíčkov, na ktorých je závislý)
- odstránenie zadaného balíčka s ošetrením závislostí (odstránenie balíčkov, ktoré sú na ňom závislé)

1 Nástroj bol testovaný len s originálnymi balíčkami dodávanými spoločnosťou SUN Microsystems, no teoreticky by mal fungovať aj s balíčkami tretích strán.

Skript bol vytvorený na účel kompenzácie nedostatkov balíčkovacieho systému výrobcu a teda na účel jednorázových inštalácií, resp. odstraňovania balíčkov a nie ako nástroj na komplexnú správu balíčkov.

Princíp činnosti

Skript je pomerne jednoduchý a pre viac informácií odporúčame prezrieť priamo jeho zdrojový kód. Systém práce s balíčkami je popísaný v kapitole Analýza, časti Balíčkovací systém OS Solaris.

6. Hodnotenia operačného systému Solaris

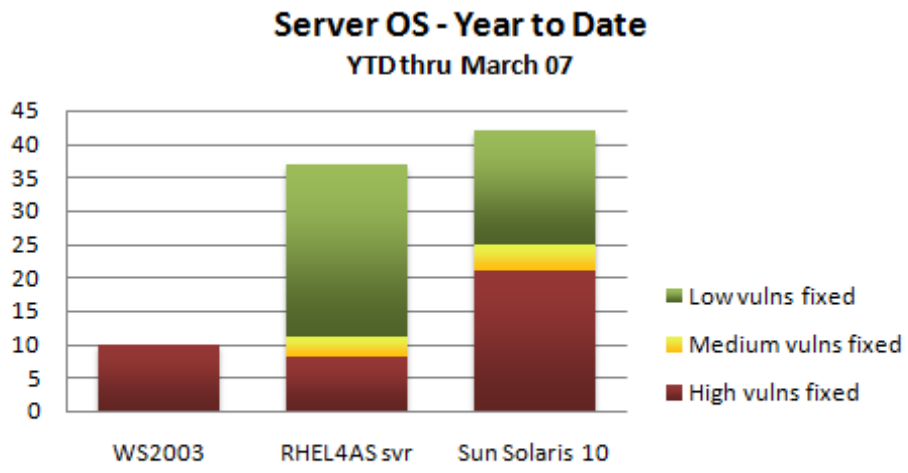
Kapitola obsahuje sumarizáciu testov a porovnaní operačného systému Solaris a porovnanie vybraných testovaných vlastností s konkurenčnými operačnými systémami. Zameriava sa na porovnanie ich bezpečnosti a rýchlosti.

Zároveň obsahuje popis a výsledky realizovaného štatistického prieskumu názorov a preferencií systémových administrátorov ohľadom operačných systémov.

6.1. Porovnanie bezpečnosti

6.1.1. Operating system vulnerability scorecard

Na webovej stránke blogs.csoonline.com sú pod názvom *Operating system vulnerability scorecard* pravidelne uverejňované sumarizácie počtov objavených bezpečnostných zraniteľností spolu s ich závažnosťou. Pre ilustráciu uvádzam porovnanie operačného systému Solaris 10 s konkurenčnými systémami RadHat Enterprise Linux 4 AS a Windows 2003 Server. Toto porovnanie sa týka len operačných systémov určených pre serverové aplikácie.



Metodika porovnania

Zraniteľnosti sú uvádzané za obdobie november 2006 - január 2007.

Zoznam zraniteľností OS Windows 2003 Server bol čerpaný z stránky výrobcu: http://www.microsoft.com/technet/security/current.aspx

Zoznam zraniteľností OS Solaris 10 bol čerpaný z stránky výrobcu http://sunsolve.sun.com/search/advsearch.do?collection=SUNALERT&type=collections&sort=date&queryKey4=%22category:security%22%20%22availability,%20security%22%20category:security&max=100

Zoznam zraniteľností OS RedHat Enterprise Linux 4 AS bol čerpaný z stránky výrobcu: http://rhn.redhat.com/errata/

Štatistiky pre Windows 2003 Server obsahujú zraniteľnosti všetkých produktov štandardne dodávaných ako súčasti Windows Server 2003, vrátane prehliadača MS Internet Explorer.

Štatistiky Red Hat Enterprise Linux 4 AS (rhel4as) obsahujú len zraniteľnosti balíčkov, ktoré tvoria minimálnu požadovanú skupinu nástrojov potrebnú na minimálnu inštaláciu operačného systému. Okrem spomenutých sú zahrnuté aj aplikácie potrebné na štandardný beh sieťových služieb ako sú file server, print server, network server a základný web server.

Štatistiky Solaris 10 obsahujú všetky zraniteľnosti, ktoré potvrdila spoločnosť SUN Microsystems, ako zraniteľnosti operačného systému Solaris 10.

Zdroj: March 2007 - Operating System Vulnerability Scorecard

http://blogs.csoonline.com/march_2007_operating_system_vulnerability_scorecard

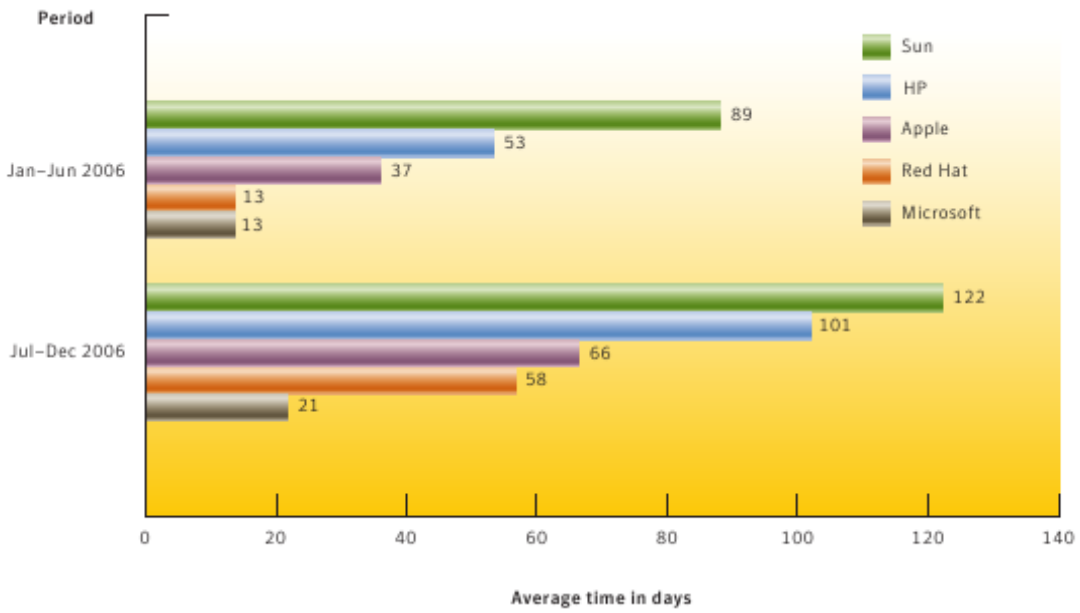
Poznámka: Všetky odkazy na webové stránky v tejto kapitole boli aktualizované ku dňu 8. máj 2007.

6.1.2. Symantec Internet Security Threat Report

Spoločnosť Symantec zameriavajúca sa na vývoj bezpečnostných aplikácií v pravidelných správach vydáva štatistiky opisujúce vývoj bezpečnosti Internetu za posledné skúmané obdobie. Tieto správy sa nazývajú *Internet security threat report*.

Čas vývoja bezpečnostných záplat

Ich súčasťou *Internet security threat report* je aj meranie priemerného času, ktorý jednotlivým výrobcom operačných systémov trvá vytvorenie bezpečnostnej záplaty (patch) pre danú zraniteľnosť. Výsledky testu za rok 2006 sú ilustrované v nasledujúcom grafe. Priemer bol počítaný zo vzoriek rádovo desiatok náhodne vybraných zraniteľností všetkých závažností. Ako zdroj informácií bola použitá verejná databáza zraniteľností[1] *Bugtraq* prístupná cez portál www.securityfocus.com.



Obrázok 13: Patch development time for operating systems - priemerný čas uvoľnenia bezpečnostnej záplaty výrobcami operačných systémov

1 Podľa Internet Security Threat Report porovnávaní producenti operačných systémov priznali priemerne 36% zraniteľností a nepriznali zvyšných 64%. Do databázy Bugtraq sa dostávajú zraniteľnosti od používateľov a nie od výrobcov softvéru. Nie všetky spomenuté zraniteľnosti výrobcovia akceptujú.

Zdroj: Symantec's 11th Internet Security Threat Report, Trends for July - December 06, Volume XI, Published March 2007.

6.1.3. Secunia.com

Na portáli **secunia.com**, ktorý sa špecializuje na bezpečnosť informačných technológií je možné nájsť priebežne aktualizovanú stránku zobrazujúcu aktuálny stav bezpečnosti OS Solaris na základe množstva a závažnosti ošetrených a neošetrených zraniteľností v danom OS. Presnejšie informácie je možné nájsť na stránke[1] http://secunia.com/product/4813/?task=statistics_2007.

6.1.4. IBM Internet Security Systems X-Force Trend Statistics

Podobnú správu ako *Symantec Internet security threat report* vydáva aj spoločnosť IBM (za bývalú ISS) pod názvom *IBM Internet Security Systems X-Force Trend Statistics*. Táto[2] ukazuje zoznam desiatich najväčších producentov softvéru z hľadiska ich percentuálneho podielu na celkovom počte známych bezpečnostných zraniteľností.

Vendor	Percentage of 2006
Microsoft Corporation	3.1%
Oracle Corporation	2.1%
Apple Computer, Inc.	1.9%
Mozilla Corporation	1.4%
IBM	1.2%
Linux Kernel Organization, Inc.	1.2%
Sun Microsystems, Inc.	1.0%
Cisco Systems, Inc.	0.9%
Hewlett-Packard	0.6%
Adobe Systems Incorporated	0.4%

Obrázok 14: Percentuálny podiel producentov softvéru na počte bezpečnostných zraniteľností za rok 2006 podľa *IBM Internet Security Systems X-Force 2006 Trend Statistics*

Celú správu je možné nájsť v [16].

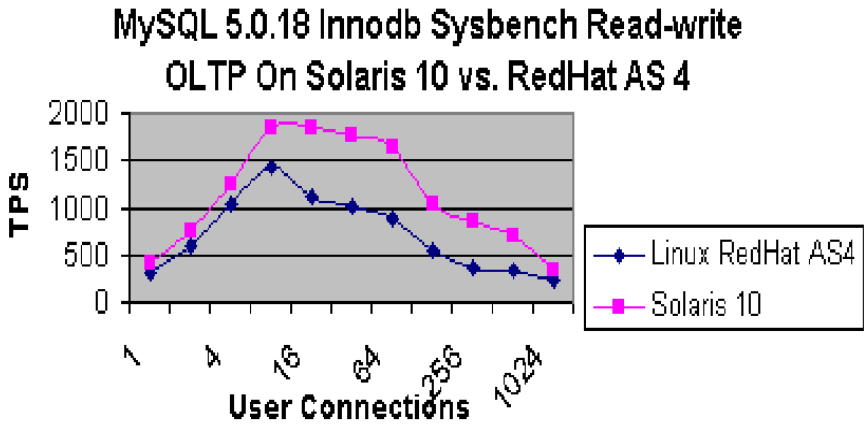
1 Odkaz je platný ku dňu 15. apríl 2007.

2 za rok 2006 na strane 11

6.2. Porovnanie rýchlosti

6.2.1. MySQL Innodb SysBench OLTP Benchmark Test

Je príkladom výkonnostný so simulovanou sieťovou prevádzkou so zameraním na výkon databázovej aplikácie MySQL so špecifickými nastaveniami. Na porovnávaných operačných systémoch Red Hat Enterprise Linux 4 AS4 a Sun Solaris 10 bola nainštalovaná databáza MySQL a s pomocou softvérového testovacieho nástroja SysBench boli generované požiadavky na databázu.



Obrázok 15: Porovnanie počtu obslúžených databázových transakcií za sekundu v OS Solaris a Red Hat Linux. Hodnoty sú vynesené pre pre rôzne počty sieťových spojení.

6.3. Vlastné meranie rýchlosti

Nástrojom *benchmark* implementovaným v rámci tejto diplomovej práce boli odmerané rýchlosti vykonávania základných operácií vybraných operačných systémov.

6.3.1. Porovnávané operačné systémy

Na porovnávací test boli okrem systému Solaris použití zástupcovia najrozšírenejších serverových operačných systémov. Pri výbere konkrétnej verzie

bola snaha o čo najaktuálnejšie a najnovšie ponúkané produkty. V porovnávacích testoch boli použité operačné systémy[1]:

1. Red Hat Enterprise Linux 4
2. Solaris 10 11/2006
3. FreeBDS 6.2

Pri testoch všetkých spomenutých systémov bol použitý výrobcom dodávaný nainštalovaný obraz pre simulátor virtuálneho počítača (vmware image).

6.3.2. Testovacie prostredie

Ako testovacie prostredie bol použitý softvér určený na simuláciu fyzického počítača – vmware virtual machine. Jedná sa o neutrálny hardvér, ktorého podporu deklarovali výrobcovia všetkých testovaných systémov. Umožňuje stanoviť zhodné podmienky testov bez ohľadu na ďalšie faktory.

Podmienky testov

- pri všetkých testoch bol použitý ten istý hardvér
- v priebehu testov ani medzi nimi neboli vykonávané žiadne zmeny hostiteľského operačného systému, simulátoru virtuálneho počítača ani súvisiacich aplikácií
- medzi jednotlivými testami boli virtuálne počítače reštartované aby boli zachované rovnaké štartovacie podmienky
- pri všetkých meraných systémoch bol testovací program najprv niekoľkokrát spustený bez záznamu výsledkov, až potom nasledovalo samotné meranie
- nástroj benchmark bol na všetkých systémoch spustený s prednastavenými parametrami v stochastickom móde

Parametre hostiteľského prostredia

Hostiteľský hardvér:	- AMD Athlon 64 X2 Dual Core Processor 3600+ - Základná doska Asus M2N8-VMX

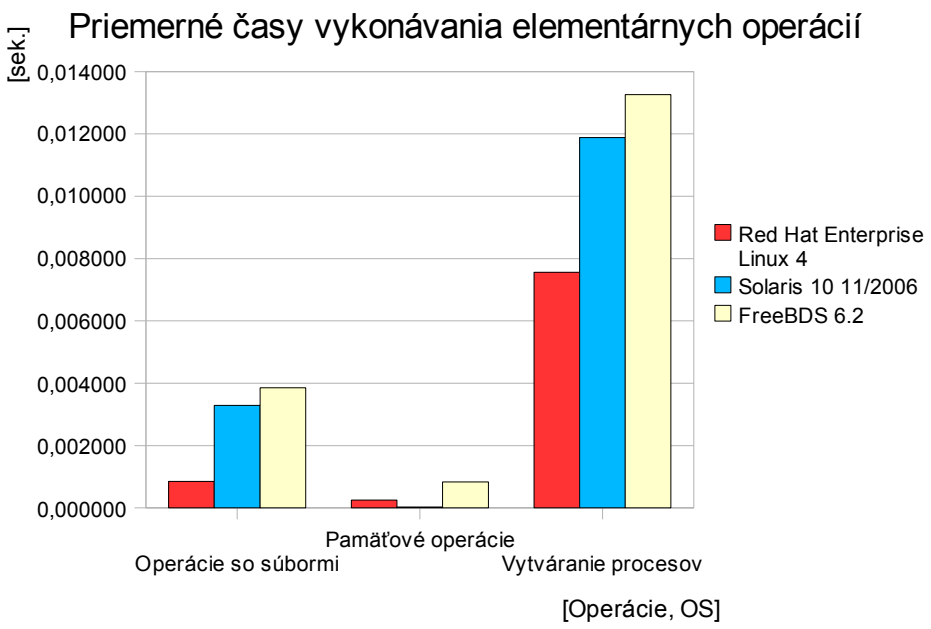
1 Pôvodne sa mal testu zúčastniť aj operačný systém Windows Server 2003 Enterprise Edition, no z dôvodu provlémovej implementácie mechanizmov na presné meranie krátkych časov a súvisiacej potenciálnej straty relevancie výsledkov bol z testu vylúčený.

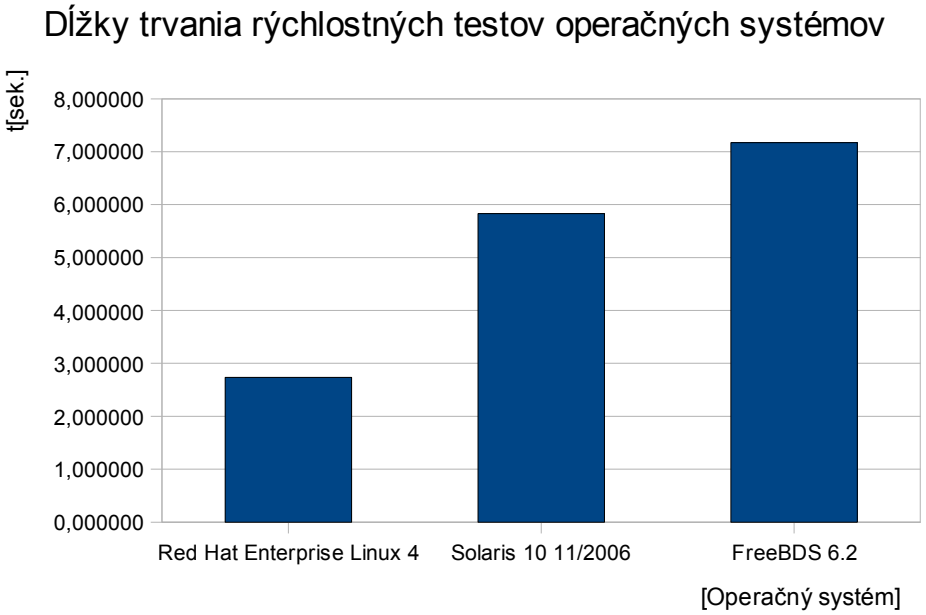
	- Sieťová karta 1000Mbps, integrovaná na základnej doske - RAM 1536MB DDRAM2 PC533 Kingston CL4
Hostiteľský operačný systém:	- Linux 64 bit, jadro 2.6.23
Simulátor:	- VMware Workstation 6.0.2 build-59824 for Linux
Virtuálny hardvér:	- VMware virtual machine - 512MB RAM

6.3.3. Výsledky testov

Rýchlosti vykonávania základných typov operácií sú zhrnuté v nasledujúcej tabuľke a graficky znázornené na nasledovných grafoch:

Popis	**Red Hat Enterprise Linux 4**		**Solaris 10 11/2006**		**FreeBDS 6.2**	
	Tsum [sek.]	T/op [sek.]	Tsum [sek.]	T/op [sek.]	Tsum [sek.]	T/op [sek.]
Operácie so súbormi	**0,864026**	**0,000853**	**3,389417**	**0,003291**	**3,709415**	**0,003856**
Vytváranie	0,023082	0,000124	0,055282	0,000305	0,371410	0,002172
Zápis	0,301904	0,001973	1,530397	0,008550	1,137727	0,008619
Dopĺňane na koniec	0,351008	0,002017	1,210160	0,007119	1,410114	0,008704
Otváranie a zatváranie	0,006745	0,000042	0,027102	0,000179	0,012063	0,000077
Čítanie	0,132861	0,000800	0,336673	0,001902	0,441320	0,002452
Odstraňovanie	0,048426	0,000282	0,229803	0,001336	0,336781	0,002092
Pamäťové operácie	**0,208367**	**0,000251**	**0,023300**	**0,000028**	**0,688489**	**0,000830**
Alokácia	0,007063	0,000040	0,005371	0,000032	0,003935	0,000022
Zápis	0,165602	0,000925	0,008095	0,000048	0,348519	0,002038
Realokácia	0,017391	0,000106	0,007511	0,000044	0,322541	0,002224
Čítanie	0,000650	0,000004	0,001483	0,000009	0,001584	0,000010
Uvoľňovanie	0,017661	0,000112	0,000840	0,000005	0,011910	0,000072
Vytváranie procesov	**1,665274**	**0,007562**	**2,416488**	**0,011882**	**2,772697**	**0,013264**
Fork	0,281364	0,001599	0,551073	0,003062	0,610691	0,003724
System	1,383910	0,007562	1,865415	0,011882	2,162006	0,013264
Celková dĺžka testu	**2,737667**	**0,001212**	**5,829205**	**0,002617**	**7,170601**	**0,003355**





Obrázok 17: Dĺžky trvania rýchlostných testov operačných systémov.

Zhodnotenie merania

Meranie prebehlo bez badateľných komplikácií. Pri viacerých opakovaniach rovnakého merania vyšli podobné výsledky. Rozdiely boli pravdepodobne zapríčinené aktivitami iných bežiacich procesov, prípadne činnosťou hostiteľského počítača.

6.4. Názory používateľov na operačný systém Solaris

Pre ilustráciu spomínaných vlastností uvádzame v nasledujúcej tabuľke typicky sa vyskytujúce názory používateľov na operačný systém Solaris, prípadne subjektívne porovnania s inými operačnými systémami. V prvom stĺpci je uvádzané, či sa jedná vo všeobecnosti o kladný, alebo záporný názor.

-	Solaris myslim ze bohuzel na dnesni home scene nema misto.
+	je to velmi dobry kus pro vetsi firmy ci nake jednosmerne idey v it
-	u me takhle spis vyhravaji veci jako suse linux cio mandriva linux a oba ve verzi x64.. pouzivam oboje
-	nevyhoda je hlavni v tom ze je prilis provazany s javou a tim silena tezkopadnost a navaznost na okolni kusy v lan ci world it
-/+	Je X verzii Linuxu zdarma /niekte su na tom s bezpecnostou celkom dobre/. Dalej ak uz platene, tak su tu platene Linux verzie SuSe, RedHat atd. Pripadne komercne UNIXy. Alebo BSD verzie - FreeBSD, NetBSD, ci bezpecne OpenBSD. Je tu toho jednoducho vela.
+	Osobne by som tento system povazoval za kompromis medzi takymi unixovymi systemami ako linux, ktore nie su pre bezneho uzivatela kvoli zlozitosti velmi priatelske, a drahymi a user-friendly-ako-sa-len-da windowsami.
+	OS Solaris 10 je prepracovanym produktom spolocnosti Sun, od ktorej zrejme vsetci windowsaci pouzivaju javu a rozhodne nejde o nejaky experiment, ale plnohodnotny funkcny produkt patriaci do svetovej elity.
+	Jeho jasnou vyhodou je, ze je momentalne pristupny na volne stiahnutie pre osobne ucely
-	nevyhoda pre tych s pomalsim internetom bude, ze ide asi o 1.5 GB archiv rozdeleny na tri baliky
+/-	Pred stiahnutim sa musite zaregistrovat, ale nejaky rapidny narast spamu som po registracii nepostrehol)
+	vela virusov na tento system pisanych nie je
+	system je vysoko stabilny
+/-	ja osobne som sa s problemom s podporou hardware nestretol, ale treba povedat, ze nejaky neznamy hardware bude zjavne problem nainstalovat, pretoze nie kazda spolocnost stavia ovladace s podporou OS Solaris
+	process a user rights management, pomerne dobre prepracovany
+	Co sa tyka antivirov, na solarise funguje F-PROT, Vexira, InterScan for Sendmail od Trend Micro, eTrust AntiVirus, Joe Fox, Sophos a udajne aj Norton Antivirus ma verziu pre tento OS
+	Sun chce prosadit svůj Solaris na úkor Linuxu na trhu platformy x86. Chce se zaměřit zejména na bezpečnost a síťové funkce. Chce nabídnout support 24/7 za poloviční cenu než nabízí Red Hat

Zdrojmi uvádzaných názorov sú odborné diskusie na stránkach[1]:

- http://www.viry.cz/forum/viewtopic.php?t=33440&sid=82aded7ec98291ecf1648be590b98f33
- http://linuxos.sk/?show=forum&id=3894
- http://www.freebsd.cz/listserv/archive/users-l/2005-July/013524.html

6.5. Prieskum názorov systémových administrátorov

V rámci tejto diplomovej práce bol realizovaný štatistický prieskum názorov a preferencií profesionálnych systémových administrátorov na záležitosti týkajúce sa operačných systémov. Táto kapitola prináša popis a vyhodnotenie realizovaného prieskumu.

6.5.1. Priebeh prieskumu

Po vytvorení otázok bola zvolená voľne dostupná bezplatná webová služba PollDaddy, do ktorej boli navrhnuté otázky zadané. Z dôvodu jednoduchosti dotazníka pre respondentov pre bola snaha o čo najväčšie využitie tzv. klikacích ovládacích prvkov.

Respondenti boli individuálne vybratí a prostredníctvom elektronickej pošty individuálne oslovení so žiadosťou o pomoc pri prieskume. Zároveň boli požiadaní o preposlanie žiadosti aj ďalším kolegom schopným poskytnúť relevantné odpovede. Jedinou podmienkou kladenou na respondentov je ich profesionalita v sfére správy operačných systémov. Neboli teda napríklad oslovované osoby, ktoré sa len učia spravovať servery.

Reakcie respondentov na test boli pomerne veľmi rýchle a pozitívne. Subjektívne možno povedať, že ich odpovede boli relevantné (napríklad neboli volené náhodné odpovede a pod.) a zodpovedali očakávaniam. Viacerí respondenti prejavili záujem o výsledky prieskumu.

Finálne sa prieskumu zúčastnilo **32** osôb. Žiadny z respondentov nebol z prieskumu vylúčený. Prieskum bol realizovaný v období 21. November až 7. December 2007.

1 Spomenuté odkazy boli platné ku dňu 9. Apríl 2007

Vstupné požiadavky

Na prieskum boli stanovené nasledovné požiadavky:

- Otázky musia byť jednoducho formulované a ľahko pochopiteľné (môže sa používať neformálna reč)
- Musí sa jednať o profesionálnych systémových administrátorov, ktorí sa venujú, prípadne sa v pomerne krátkej minulosti venovali správe produkčných serverov ako jednej z hlavných náplní práce.
- Počet otázok nesmie byť väčší ako 10.
- Nesmie byť účelovo oslovená komunita (napríklad internetové fórum) združená okolo jedného konkrétneho operačného systému.
- Prieskum je plne anonymný.

6.5.2. Otázky a vyhodnotenie

V ďalšom sú vymenované jednotlivé testové otázky aj so štatistikami odpovede na ne. Priložené tabuľky a grafy sú priamym výstupom dotazníkovej aplikácie. Obsahujú všetky možnosti odpovedí, ktoré respondenti dostali a početnosť aj percentuálny výskyt jednotlivých odpovedí.

Poznámky k otázkam:

- Číslo pod stĺpcom "Votes" reprezentuje počet celkovo zaznamenaných hlasov v danej otázke. V mnohých otázkach (v tých, kde je počet celkovo zaznamenaných hlasov väčší ako počet hlasujúcich) bolo možné označiť aj viac možností (dať viac odpovedí).
- Žiadna z otázok nebola povinná.
- Okrem otázky č. 9 boli všetky otázky zodpovedané všetkými respondentmi.
- V otázkach, kde bolo potrebné ohodnotiť pravdivosť tvrdení (otázky č. 3, 5 a 8) mohli respondenti pri každom z tvrdení vybrať len jednu možnosť.

Otázka č. 1: Aké operačné systémy používate najčastejšie?

	Answer Text	Votes	%	
	Linux	31	38.75%	
	Windows	19	23.75%	
	Solaris	12	15.00%	
	*BSD	6	7.50%	
	HP-UX	6	7.50%	
	OSX	4	5.00%	
	Other Option...	1	1.25%	
	AIX	1	1.25%	
		80		

Tabuľka 2: Odpovede na otázku č. 1.

Poznámky:

- Položka Other Option obsahovala odpoveď "**novell**".

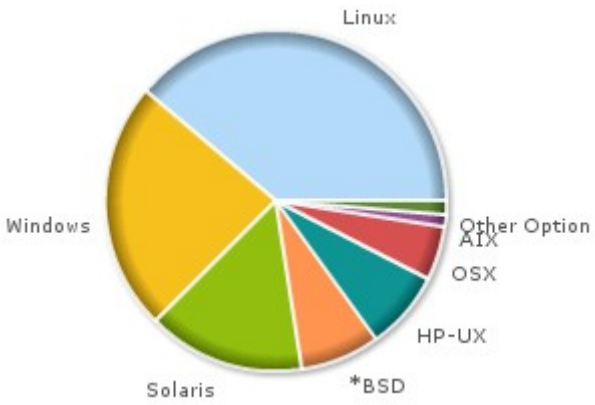


Diagram 1: Odpovede na otázku č. 1.

Otázka č. 2: Aké operačné systémy máte najradšej?

	Answer Text	Votes	%	
	Linux	27	49.09%	
	Solaris	8	14.55%	
	Windows	8	14.55%	
	*BSD	4	7.27%	
	OSX	4	7.27%	
	HP-UX	3	5.45%	
	Other Option...	1	1.82%	
	AIX	0	0.00%	
		55		

Tabuľka 3: Odpovede na otázku č. 2.

Poznámky:

- Položka Other Option obsahovala odpoveď "**novell**".

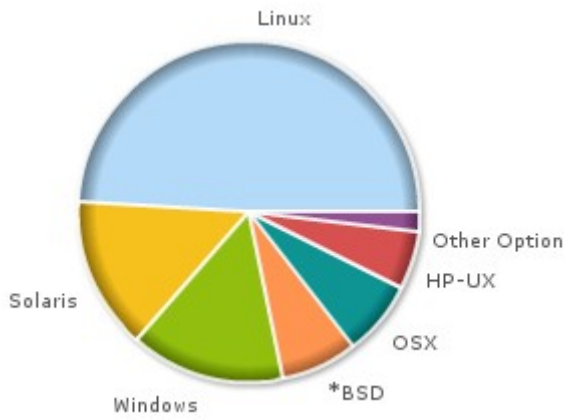


Diagram 2: Odpovede na otázku č. 2.

Otázka č. 3: Ohodnoťte dôležitosť vlastností operačných systémov!

	Naozaj veľmi dôležité	Dôležité	Tak stredne	Môže byť	Nezláleží mi na tom
dokumentácia	9 (28.13%)	17 (53.13%)	4 (12.5%)	2 (6.25%)	0 (0%)
podpora od výrobcu/ komunity	8 (25%)	17 (53.13%)	6 (18.75%)	1 (3.13%)	0 (0%)
stabilita	28 (87.5%)	3 (9.38%)	1 (3.13%)	0 (0%)	0 (0%)
cena	3 (9.38%)	2 (6.25%)	16 (50%)	7 (21.88%)	4 (12.5%)
rýchlosť (nároky na výkon hardvéru)	6 (18.75%)	19 (59.38%)	6 (18.75%)	1 (3.13%)	0 (0%)
hardvérová podpora (na čom všetkom sa to dá rozbehať)	5 (15.63%)	15 (46.88%)	9 (28.13%)	2 (6.25%)	1 (3.13%)
konfigurovateľnosť (možnosti prispôsobenia)	11 (34.38%)	16 (50%)	5 (15.63%)	0 (0%)	0 (0%)
jednoduchosť inštalácie	0 (0%)	5 (16.13%)	7 (22.58%)	9 (29.03%)	10 (32.26%)
nenáročnosť údržby	2 (6.25%)	12 (37.5%)	13 (40.63%)	2 (6.25%)	3 (9.38%)
bezpečnosť	21 (65.63%)	11 (34.38%)	0 (0%)	0 (0%)	0 (0%)

Tabuľka 4: Odpovede na otázku č. 3.

Vyhodnotenie

Pre každé tvrdenie bol počet respondentov, ktorí zvolili danú možnosť prenásobený koeficientom danej možnosti a výsledky týchto súčinov sa pre všetky možnosti sčítali. Celkové výsledky sú potom uvádzané v priložených výsledkoch. Koeficienty stĺpcov boli (v poradí) 3, 2, 1.5, 1, 0. Po prenásobení nimi a sčítaní dostávame v poradí nasledujúce výsledky:

Vlastnosť	Úspešnosť
stabilita	91,5
bezpečnosť	85
konfigurovateľnosť (možnosti prispôsobenia)	72,5
dokumentácia	69
podpora od výrobcu/komunity	68
rýchlosť (nároky na výkon hardvéru)	66
hardvérová podpora (na čom všetkom sa to dá rozbehať)	60,5
nenáročnosť údržby	51,5
cena	44
jednoduchosť inštalácie	29,5

Tabuľka 5: Vyhodnotenie odpovedí na otázku č. 3.

Otázka č. 4: Moje servery sú priemerne vyťažené na:

	Answer Text	Votes	%	
	0 - 25%	13	40.63%	
	26 - 50%	12	37.50%	
	51 - 75%	5	15.63%	
	76 - 100%	2	6.25%	
		32		

Tabuľka 6: Odpovede na otázku č. 4.

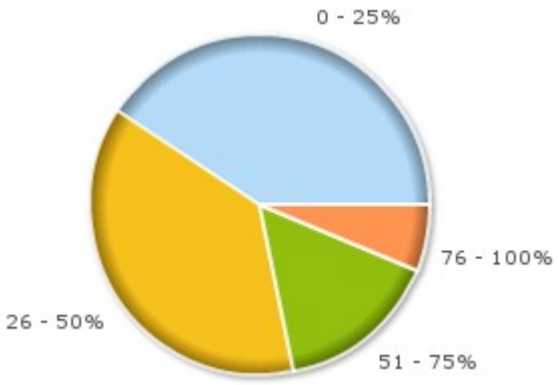


Diagram 3: Odpovede na otázku č. 4.

Otázka č. 5: Moje servery sú v špičkách vyťažené na:

	Answer Text	Votes	%	
	90 - 100%	12	37.50%	
	75 - 90%	11	34.38%	
	50 - 75%	7	21.88%	
	0 - 50%	2	6.25%	
		32		

Tabuľka 7: Odpovede na otázku č. 5.

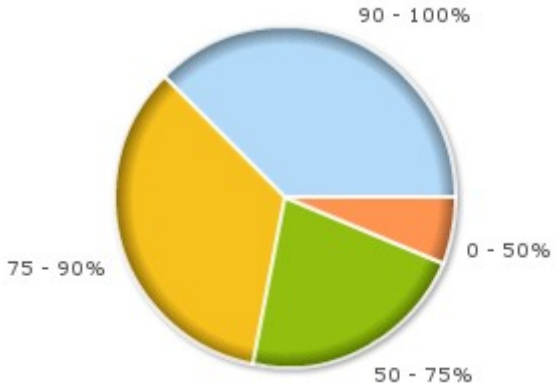


Diagram 4: Odpovede na otázku č. 5.

Otázka č. 6: Ohodnoťte pravdivosť výrokov!

	Súhlasím	Áno aj nie	Nesúhlasím
10% úbytku rýchlosti mojich serverov by neznamenalo problém.	18 (56.25%)	12 (37.5%)	2 (6.25%)
Viac služieb, ktoré poskytujú naše servery by sa mohlo kľudne zmestiť aj na na menej fyzických počítačov bez zníženia kvality služieb.	13 (40.63%)	8 (25%)	11 (34.38%)
Pomerne často používame hardverový RAID.	16 (50%)	7 (21.88%)	9 (28.13%)
Pomerne často používame softvérový RAID.	9 (28.13%)	10 (31.25%)	13 (40.63%)
Je mi jedno koľko to stojí, keď to neplatím ja.	12 (37.5%)	11 (34.38%)	9 (28.13%)
Najradšej by som mal všetkých serveroch rovnaký operačný systém.	14 (43.75%)	11 (34.38%)	7 (21.88%)
Pri nákupe serverov sa v našej spoločnosti veľmi nešetrí.	8 (25%)	12 (37.5%)	12 (37.5%)
Rád sa naučím pracovať s novým OS.	24 (75%)	7 (21.88%)	1 (3.13%)
Spotreba elektrickej energie, klimatizácia a nedostatok fyzického miesta pre servery v našej spoločnosti nie je problém.	9 (28.13%)	10 (31.25%)	13 (40.63%)
Spotreba elektrickej energie, klimatizácia a nedostatok fyzického miesta pre servery v budúcnosti asi ani nebude problém (ak nepredpokladáme zásadnú zmenu prístupu spoločnosti k týmto problémom).	9 (28.13%)	10 (31.25%)	13 (40.63%)
Preferujem radšej zaužívané postupy.	14 (43.75%)	16 (50%)	2 (6.25%)

Tabuľka 8: Odpovede na otázku č. 6.

Poznámky:

- Keďže jednotlivé tvrdenia spolu navzájom nesúvisia, túto otázku nie je potrebné vyhodnocovať tak, ako otázku č. 3.

Otázka č. 7: V našej spoločnosti používame virtualizačné technológie:

	Answer Text	Votes	%	
	free VMware	20	33.90%	
	platené VMware	15	25.42%	
	Solaris virtualizácia	8	13.56%	
	Xen	7	11.86%	
	Microsoft virtualizácia	4	6.78%	
	nepoužívame virtualizačné technológie	3	5.08%	
	Other Option...	2	3.39%	
		59		

Tabuľka 9: Odpovede na otázku č. 7.

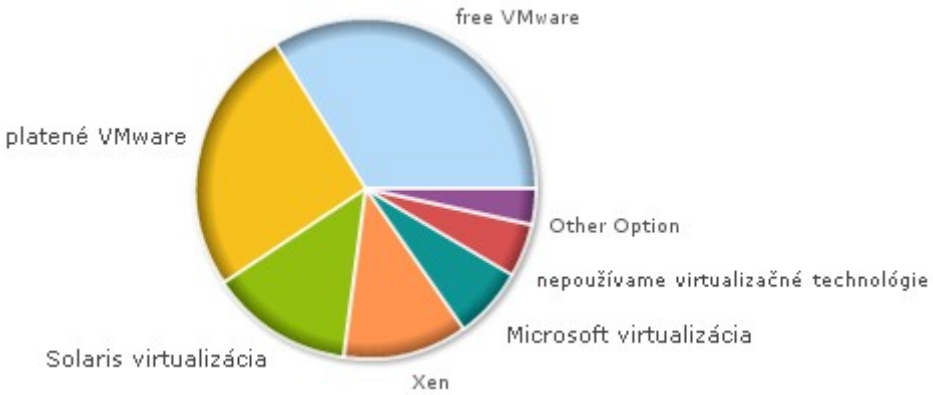


Diagram 5: Odpovede na otázku č. 7.

Poznámky:

- Položka Other Option obsahovala odpovede "**vserver**" a "**kvm, qemu**".

Otázka č. 8: Privítal by som pokročilé možnosti práce s diskami/partíciami ako napr.:

	Určite	Možno	Nie
okamžitá zmena veľkostí partícií za behu	22 (70.97%)	8 (25.81%)	1 (3.23%)
pridávanie a odstraňovanie diskov/partícií za behu	21 (70%)	9 (30%)	0 (0%)
snapshot (vytvorenie image) disku/partície za behu	23 (74.19%)	8 (25.81%)	0 (0%)
pokročilé ACL	16 (51.61%)	14 (45.16%)	1 (3.23%)

Tabuľka 10: Odpovede na otázku č. 8.

Otázka č. 9: Plat systémového administrátora na Slovensku je priemerne:

Pod znením otázky bolo upresnenie: "Hrubá mzda v Skk/mesiac pri plnom úväzku"

Pri tejto otázke boli respondenti požiadaní o zadanie konkrétneho čísla. Otázka je účelovo položená tak, aby sa nepýtala konkrétne na plat daného respondenta čo by v ňom pravdepodobne vyvolalo zábrany odpovedať. Z 32 respondentov na túto otázku odpovedalo 28. Štatistické údaje vypočítané z odpovedí sú:

- Minimálna hodnota: 25000 Sk
- **Priemerná hodnota: 37111 Sk**
- Maximálna hodnota: 60000 Sk

Cieľom tejto otázky bolo vypočítanie priemerných nákladov zamestnávateľa na hodinu práce systémového administrátora. Tento výpočet sa nachádza v časti *Náklady na prácu systémového administrátora* tejto kapitoly.

Otázka č. 10: Rozbehať jeden server mi trvá priemerne:

	Answer Text	Votes	%	
	1 - 4 hod.	16	50.00%	
	0.5 - 1 deň	8	25.00%	
	1 - 3 dni	6	18.75%	
	3 - 7 dní	1	3.13%	
	menej než 1 hod.	1	3.13%	
	viac než 7 dní	0	0.00%	
		32		

Tabuľka 11: Odpovede na otázku č. 10.

Viacerým respondentom nebol jasný presný význam pojmu "rozbehať". Bolo tým myslené nainštalovanie operačného systému a inicializačná konfigurácia do takého stavu, v akom je server pripravený na inštaláciu služieb, no okrem nich už žiadne ďalšie zásahy nepotrebuje. Týka sa to najmä činností ako: zabezpečenie aktualizácie, zabezpečenie súladu s firemnými bezpečnostnými požiadavkami, zálohovanie napájania a nastavenie automatického vypnutia servera pri výpadku napájania a pod. Toto však nebolo upresnené, čo spôsobilo oprávnene rozdielne pochopenie otázky medzi disponentmi.

6.5.3. Náklady na prácu systémového adminsitrátora

Pre zaujímavosť uvádzam výpočet nákladov na jednu hodinu práce systémového administrátora. Prieskum bol realizovaný vzhľadom na finančné pomery v Slovenskej Republike. Pri výpočte vychádzame z priemernej hrubej mesačnej mzdy 37111 Sk.

Na výpočet nákladov bola použitá webová služba dostupná na adrese[1] http://www.financnik.sk/financie.php?did=155

1 Odkaz bol platný dňa 7. decembra 2007.

Obrázok 18: Výstup programu na výpočet celkových ročných

Hrubá mesačná mzda spolu:	**37 111,00**
Príspevky na DDP vám zamestnávateľ hradí v (ak prispieva)[1]:	%
Hodnota príspevku vášho zamestnávateľa (SKK alebo %):	**0,00**

Odvody do poisťovní

		do 30.9.2007	od 1.10.2007
Starobné poistenie: (+)	14,00%	5 196,00	5 196,00
Nemocenské poistenie: (+)	1,40%	394,00	394,00
Poistenie v nezamestnanosti: (+)	1,00%	372,00	372,00
Invalidné poistenie: (+)	3,00%	1 114,00	1 114,00
Úrazové poistenie: (+)	0,80%	297,00	297,00
Garančné poistenie: (+)	0,25%	71,00	71,00
Rezervný fond solidarity: (+)	4,75%	1 763,00	1 763,00
Zdravotné poistenie: (+)	10,00%	3 712,00	3 712,00
Odvody spolu:	35,20%	12 919,00	12 919,00

Celkové mesačné náklady na zamestnanca:	50 030,00	(+0) **50 030,00**
Celkové ročné náklady na zamestnanca:	600 360,00	(+0) **600 360,00**
Celkové mesačné náklady na zamestnanca v EUR*:	1 411,59	(+0) **1 411,59**
Celkové ročné náklady na zamestnanca v EUR*:	16 939,03	(+0) **16 939,03**

* Prepočítané kurzom centrálnej parity koruny voči EUR (1 euro = 35,4424 Sk).

nákladov na zamestnanca.

S prihliadnutím na 160 hodín odpracovaných za jeden mesiac a 20 dňoch platenej dovolenky ročne, dostávame náklady na jednu reálne odpracovanú hodinu systémového administrátora vo výške **341.11 Sk**.

Do výpočtu neboli zahrnuté náklady súvisiace s odmenami, nadčasmi, pohotovosťou, preplácaním cestovných nákladov, nákladov na stravu, školeniami a ďalšími sociálnymi výhodami poskytovanými spoločnosťou.

6.6. Vlastné skúsenosti

Kapitola obsahuje popis činností vykonávaných za účelom odskúšania reálnej prevádzky operačného systému Solaris a sumarizáciu pozitívnych aj negatívnych skúseností práce s týmto systémom.

Operačný systém bol testovaný na testovacom serveri, ktorý zároveň poskytoval aj reálne služby počas obdobia viac než dvoch rokov. Počas tohto obdobia bol používaný ako univerzálny zdroj s možnosťou inštalácie prakticky ľubovoľných

služieb. Hneď v úvode bol prispôsobený na prostredie podobné OS Linux. Služby boli inštalované z balíčkov výrobcu, balíčkov komunity *blastwave* aj balíčkov tretích strán. Server zároveň slúžil na skúšanie novo naučených postupov špecifických pre OS Solaris. Keďže sa s ním pracovalo pomerne veľmi pohodlne, bol často využívaný, ale používateľom chýbali potrebné vedomosti, takže došlo k niekoľkým vážnym zásahom, ktoré ale neovplyvnili jeho schopnosti poskytovania sieťových služieb. Po dvoch rokoch bol reinštalovaný a nanovo spustený už v prostredí zóny (ako virtuálny server) a s využitím súborového systému ZFS. Bol nainštalovaný aj ďalší podobný server založený na architektúre SPARC, tento však používaný výlučne na produkčné účely.

6.6.1. Realizované operácie

Skúsenosti vychádzajú z nasledovných realizovaný úkonov:

- inštalácia, konfigurácia a reálna prevádzka OS Solaris 10 na platformách x86 aj SPARC.
- konfigurácia a správa zón (virtuálnych serverov)
- konfigurácia a správa súborového systému ZFS
- inštalácia, konfigurácia a reálna prevádzka služieb ako:
    - MySQL
    - PostrgreSQL
    - Apache + PHP/Python
    - FTP
    - Samba
    - OpenVPN
- množstvo webových služieb
- vývoj vlastných aplikácií
- konfigurácia sieťových služieb
- ďalšie úkony súvisiace so správou serverov

Pohľad na jednotlivé aspekty OS

Stabilita

Počas celej prevádzky nebol zaznamenaný pád ani tzv. zmrznutie programu spôsobené operačným systémom. Po náhlych reštartoch vždy spoľahlivo naštartovala služba ssh a niektoré ďalšie služby, no od začiatku až do súčasnosti sústavne nastávajú problémy so štartovacou sekvenciou – automatickým štartom služieb.

Inštalácia

Inštalácia OS Solaris 10 nie je úplne dotiahnutá a neposkytuje napríklad prednastavené sady balíčkov optimálne pre beh sieťového servera. Množstvo balíčkov je potrebné ručne dopĺňať, prípadne sa uspokojiť s prítomnosťou veľkého množstva nepotrebných nástrojov a služieb a patričnej strate diskovej kapacity. Ďalšie problémy nastávajú najmä v prípade, ak používateľ nechce používať na správu systému grafické používateľské rozhranie. Prostredie príkazového riadku bolo potrebné ešte značne upravovať aby bola možná pohodlná práca so systémom.

Bezpečnosť

Aj napriek vyše roka nefunkčnému[1] balíčkovaciemu a aktualizačnému systému nebol zaznamenaný žiadny úspešný útok na systém. Bezpečnostné skeny tiež neindikovali žiadne závažné zraniteľnosti.

Rýchlosť

Subjektívne server vykazuje rovnakú alebo vyššiu rýchlosť v porovnaní OS Linux. Server nikdy nevykazoval známky bezdôvodného preťaženia.

Konfigurovateľnosť

Možnosti konfigurácie sú veľmi vysoké, no v porovnaní s OS Linux mierne obmedzenejšie. Vďaka funkčnosti štandardných knižníc OS Linux, kompilátora gcc a súvisiacich nástrojov bolo možné sfunkčniť skoro všetky aplikácie bežne dostupné na OS Linux. Rovnako bolo možné prispôsobiť pracovné prostredie.

1 Táto nefunkčnosť bola spôsobená vážnym zásahom (chybou) používateľa.

6.6.2. Pozitívne hodnotené aspekty OS Solaris

Subjektívne možno pozitívne hodnotiť tieto aspekty:

- vysoká stabilita a bezpečnosť (aj napriek nešetrnému zaobchádzaniu, napr. mnoho tzv. tvrdých reštartov)
- vysoká kompatibilita s OS Linux
- veľké množstvo softvérových aplikácií fungujúcich pod OS Solaris 10
- veľký progres
    - OpenSolaris (rýchlo pribúdajúce používateľsky pohodlné riešenia)
    - rýchlo rastúci počet priaznivcov
- veľmi veľká podpora výrobcu
- otvorenosť zdrojových kódov
- bezplatná dostupnosť
- vysoká konfigurovateľnosť
- nadštandardné vlastnosti
- mimoriadna jednoduchosť niektorých pokročilých komponentov
    - ovládanie zfs
    - správa zón
    - správa štartovacích skriptov (služieb)

6.6.3. Negatívne hodnotené aspekty OS Solaris

Subjektívne možno negatívne hodnotiť tieto aspekty:

- nevhodná koexistencia SUN nástrojov a opensource nástrojov
- nedotiahnutý inštalačný proces
- zle navrhnuté skupiny balíčkov
    - chýbajú užitočné aplikácie (mc, man, nmap, nc, tar -z/-j)
    - nainštalujú sa nepotrebné implicitné služby (napr. telnet server, print server a pod.)
- prednastavené prostredie je nepohodlné pre používateľa
    - nepohodlný default shell (nefungujú tlačidlá ako Home, End, Delete, šípky, funkčné klávesy a pod.)

    - nástroje nie sú implicitne farebné (napr. ls sa ani farebne nastaviť nedá)
    - premenná prostredia PATH nezahŕňa väčšinu užitočných nástrojov
    - chýba domovský adresár používateľa root
- zložitá adresárová štruktúra
- chybový mechanizmus štartovania služieb (niektoré služby vôbec nenabehnú)
- nedotiahnutý jednotný balíčkovací a aktualizačný systém
    - balíčkovací systém nerieši závislosti
    - použiteľný opensourcový nástroj je potrebné ručne doinštalovať a konfigurovať
    - chýba jednotný webový zdroj aktualizácií
    - aktualizácie nie sú dostatočne časté
    - chybový SUN aktualizačný systém
- *unstable* balíčky komunity Blastwave sú čerstvé bez dostatočného testovania, *stable* balíčky nie sú dostatočne rýchlo aktualizované, pričom sa odporúča inštalovať práve tieto
- zložité vybrané operácie
- často neprofesionálne naportovaný, prípadne vytvorený opensource softvér

6.6.4. Komentár

nedostatky sú spôsobené najmä príchodom do opensource sféry, kde platia iné pravidlá a pomerne konzervatívny systém Solaris sa im ťažko prispôsobuje. Výrobca je postavený pred otázku miery konzervativizmu, ktorú chce u produkčnej verzie zachovať.

mimoriadne pozitívne je možné hodnotiť progres, ktorý pri zachovaní súčasnej rýchlosti pravdepodobne do niekoľkých rokov vyrieši väčšinu spomínaných problémov a Solaris bude kombinovať používateľskú pohodlnosť OS Linux a stabilitu konzervatívneho OS Solaris.

7. Záver

Kapitola vymenúva práce zrealizované na projekte a uvádza zhodnotenie ich výsledkov.

7.1. Práce zrealizované v rámci projektu

V rámci tohto projektu bolo zrealizované:

- inštalácia, konfigurácia a reálna prevádzka viacerých fyzických aj virtuálnych serverov s OS Solaris 10 (na platforme x86 aj SPARC) v prostredí počítačovej siete,
- teoretické preskúmanie operačného systému ako celku
- teoretická analýza jeho vybraných vlastností:
    - Solaris Kontajnery
    - Solaris Zóny
    - Zettabyte File System
    - Balíčkovací systém
    - Bezpečnostné prvky
- praktická analýza systému ako celku a jeho vybraných aspektov:
    - konfigurovateľnosť a komfort obsluhy
    - stabilita
    - bezpečnosť
    - systém aktualizácií a záplat
    - systém správy služieb
    - sieťové nastavenia
    - virtualizácia
- vypracovanie metodiky hodnotenia operačných systémov,
- zozbieranie a sumarizácia existujúcich testov a hodnotení skúmaného operačného systému,
- štatistický prieskum názorov a preferencií systémových administrátorov ohľadom operačných systémov,

- vypracovanie používateľských návodov popisujúcich základné a vybrané pokročilé konfigurácie operačného systému Solaris,
- implementácia sady skriptov a nastavení zjednodušujúcich správu operačného systému spolu s nástrojom na ich automatickú inštaláciu,
- implementácia skriptu na ošetrovanie závislostí pri inštalácii a odstraňovaní originálnych balíčkov,
- implementácia aplikácie *benchmark* určenej na kvantitatívne meranie výkonu operačného systému,

Informácie zhromaždené v časti analýza tvoria podstatnú časť produktu tohto projektu. Ich získaniu a sformulovaniu do súčasnej podoby bolo (okrem vlastnej práce s OS Solaris) venované podstatné množstvo času.

7.2. Zhodnotenie

Projekt sa zameriava na skúmanie operačného systému Solaris 10 a jeho vybraných pokročilých vlastností v kontexte reálnej prevádzky a súčasných požiadaviek kladených na produkčné systémy. Okrem analýzy samotného operačného systému sa práca venuje aj analýze súčasného prostredia a nárokov kladených na operačné systémy. Pokúša sa analyzovať a sumarizovať zaujímavé informácie zo súvisiacich oblastí. Operačný systém Solaris je porovnávaný s konkurenčnými systémami a najmä z výsledkov týchto porovnaní vychádzajú jeho pozitívne alebo negatívne hodnotenia.

Veľmi dôležitý v tejto oblasti je personálny aspekt. V súčasnosti sú operačné systémy vyberané a hodnotené najmä prostredníctvom subjektívnych názorov ich používateľov, prípadne filozofie spoločnosti. V praxi sú znalosti a skúsenosti konkrétnych správcov serverov spolu s konzistenciou rozloženia operačných systémov spoločnosti kľúčové pri výbere operačného systému. Je veľmi obtiažne nájsť kvantitatívne ukazovatele objektívne porovnávajúce operačné systémy.

V časti analýza tohto dokumentu je predstavený OS Solaris ako celok a konkrétnejšie sú preskúmané jeho vybrané vlastnosti. Zároveň je navrhnutá všeobecne použiteľná metodika hodnotenia a porovnávania operačných systémov. Aspekty ich hodnotenia sú vymenované v úvode dokumentu. Pri generovaní textových materiálov popisujúcich vlastnosti skúmaného systému a princípy ich činnosti sa vychádzalo najmä z oficiálnych materiálov výrobcu.

Práca sa ďalej venuje testovaniu konkrétneho systému v reálnej prevádzke. Táto sa začala viac než rok pred začiatkom písania tejto práce. Počas nej boli otestované mnohé vlastnosti a vykonané rôzne úkony nad skúmaným systémom. Server, na ktorom sa experimenty realizovali bol zároveň zapojený v reálnej prevádzke. Počas mnohých, často necitlivo realizovaných, experimentov preukázal vysokú mieru stability a bezpečnosti.

Hodnotenie silných a slabých stránok OS Solaris približne kopíruje poradie preferencií systémových administrátorov kladených na operačné systémy zistených v realizovanom štatistickom prieskume. Osobné skúsenosti s prácou s týmto systémom ukázali aj viaceré nedostatky, najmä v oblasti správy systému a používateľského komfortu. Ako kompenzácia týchto nedostatkov bola vytvorená sada post-inštalačných skriptov, ktoré správcovi systému automaticky nastavia prostredie na pohodlnú prácu a realizujú ďalšie dôležité iniciálne úkony potrebné pre začiatok prevádzky sieťového servera.

OS Solaris s verziou 10 priniesol pokročilé a zaujímavé vlastnosti vďaka ktorým sa jednoznačne zaraďuje medzi svetovú špičku v oblasti serverových operačných systémov. Aj napriek množstvu nájdených nedostatkov je finálne hodnotenie operačného systému pozitívne.

Jedným z najväčších prínosov tejto práce sú nadobudnuté osobné skúsenosti autora a získané nové informácie zo skúmaných oblastí. Práca priniesla a objasnila súvislosti mnohých zaujímavých informácií. Vytvorené softvérové produkty a návody už boli uverejnené na internete a postupne budú publikované aj ďalšie textové výstupy tejto práce. Niektoré ňou iniciované projekty zostávajú aj naďalej otvorené a budú dopĺňané tak, aby priniesli úžitok najmä širokej komunite systémových administrátorov a podporili vzdelávanie v oblasti operačných systémov.

Použitá literatúra

[1] OpenSolaris.org Official center for Sun Microsystems' OpenSolaris open-source project. http://www.opensolaris.org september 2006

[2] Anil Buntwal Somayaji: Operating System Stability and Security through Process Homeostasis. Dissertation paper, The University of New Mexico, 2002 http://www.scs.carleton.ca/~soma/pubs/soma-diss.pdf

[3] Wikipedia, the free encyclopedia, http://www.wikipedia.org

[4] Official Sun documentation page, http://docs.sun.com

[5] Sun Microsystems, official page, http://www.sun.com

[6] BigAdmin System Administrator Resources and Community, http://www.sun.com/bigadmin

[7] Menno Lageman, Sun Client Solutions, Sun BluePrints™, 2005, Solaris Containers - What They Are and How to Use Them, Part No. 819-2679-10, Revision 1.0, 5/26/05

[8] Jeff Bonwick, Matt Ahrens, Val Henson, Mark Maybee, Mark Shellenbaum, The Zettabyte File System, http://partneradvantage.sun.com/protected/solaris10/adoptionkit/tech/zfs/zfs_overview.pdf

[9] Max Bruning, opensolaris.org, 2005, A Comparison of Solaris, Linux, and FreeBSD Kernels

[10] Sun Microsystems, A Technical White Paper, November 2004, The Solaris 10 Operating System

[11] Sun Microsystems, A Technical White Paper, September 2004, Solaris 10 Operating System - Unparalleled Security to Enable and Protect Business

[12] Sun Microsystems, A Technical Introduction, September 2004, Predictive Self-Healing in the Solaris 10 Operating System

[13] Australian pc world, IDG Communications, Buyers' Guide to Server Operating Systems

[14] Richard McDougall, Jim Mauro, Prentice Hall, 7.10.2006, ISBN-13: 9780131482098, Solaris™ Internals: Solaris 10 and OpenSolaris Kernel Architecture

[15] Symantec's 11th Internet Security Threat Report, Trends for July - December 06, Volume XI, March 2007, Symantec Corporation, 20330 Stevens Creek Blvd., Cupertino, CA 95014, http://eval.symantec.com/mktginfo/enterprise/white_papers/ent-whitepaper_internet_security_threat_report_xi_03_2007.en-us.pdf

[16] IBM Internet Security Systems X-Force 2006 Trend Statistics, Distribution: General XF-XFORCEEXECBRIEF-0107, 2007, IBM Corporation, 6303 Barfield Road, Atlanta, GA 30328, United States, http://www.iss.net/documents/whitepapers/X_Force_Exec_Brief.pdf

[17] MySQL Innodb SysBench OLTP Benchmark Test, April 2006, Sun Microsystems, http://www.sun.com/x64/docs/MySQL-sysbench-benchmark.pdf

[18] Jeff Jones, CSO – The Resource for Security Executives, March 2007 - Operating System Vulnerability Scorecard, http://blogs.csoonline.com/march_2007_operating_system_vulnerability_scorecard

[19] Peter Ronai, root.cz, Solaris Packaging - jednoduchý návod, http://www.root.cz/clanky/solaris-packaging-jednoduchy-navod/

[20] Image Packaging System (IPS) Assessment, http://wiki.updatecenter.java.net/Wiki.jsp?page=IPSAssessment

[21] dsl.sk, Revolučný filesystém ZFS štandardným v Mac OS X 10.5, http://www.dsl.sk/article.php?article=3995

[22] ZFS, Wikipedia, the free encyclopedia, http://en.wikipedia.org/wiki/ZFS

[23] SysBench: a system performance benchmark, http://sysbench.sourceforge.net/

[24] Michal Kára, root.cz, Fakta o faktech Microsoftu, http://www.root.cz/clanky/fakta-o-faktech-microsoftu/

[25] VeriTest, Microsoft Windows Server 2003 with Internet Information Services (IIS) 6.0 vs. Linux Competitive Web Server Performance Comparison,

http://download.microsoft.com/download/0/7/1/0715a190-70f5-4b0d-8ced-f9d1e046aa6a/webbench.pdf

Všetky odkazy na internetové zdroje boli aktuálne ku dňu 7. december 2007.

Zoznam príloh:

1. Návod na konfiguráciu a správu zón
2. Návod na konfiguráciu a správu súborového systému ZFS
3. Návod na aktiváciu a nastavenie sieťového filtra (firewall) a prekladu sieťových adries (NAT)
4. Návod na inštaláciu a konfiguráciu systému opensource balíčkov
5. Návod na aktualizáciu a distribúciu záplat (patch)
6. Tipy a triky
7. Návod na post-inštalačnú a konfiguračnú procedúru
8. Návod na použitie skriptu sundepend
9. Návod na použitie programu OS benchmark

Príloha č. 1. - Návod na konfiguráciu a správu zón

Príloha č. 2 - Návod na konfiguráciu a správu súborového systému ZFS

Príloha č. 3 - Návod na aktiváciu a nastavenie sieťového filtra (firewall) a prekladu sieťových adries (NAT)

Príloha č. 4 - Návod na inštaláciu a konfiguráciu systému opensource balíčkov

Príloha č. 5 - Návod na aktualizáciu a distribúciu záplat (patch)

Príloha č. 6 - Tipy a triky

Príloha č. 7 - Návod na post-inštalačnú a konfiguračnú procedúru

Príloha č. 8 - Návod na použitie skriptu sundepend

Príloha č. 9 - Návod na použitie programu OS benchmark

Elektronické médium