



Grid aprēķinu vide

teorija • metodes • aprēķini



XEN VEIKTSPĒJA

XEN VIRTUALIZĀCIJAS SISTĒMAS VEIKTSPĒJAS NOVĒRTĒŠANA

Dokumenta faila vārds:	LG-nodevums-xen-veiktspēja.doc
Aktivitāte:	2.aktivitāte "Starpprogrammatūras rīku praktiskā realizācija"
Projekta numurs:	VPD1/ERAF/CFLA/05/APK/2.5.1./000055/027
Organizācija:	Latvijas Universitātes aģentūra "Latvijas Universitātes Matemātikas un informātikas institūts"
Autori:	Leo Truksāns, Baiba Kaškina

Anotācija:

Xen virtuālās mašīnas ir izmantojamas lielākajai daļai serveru un skaitļošanas uzdevumu. Šajā nodevumā aprakstīti Latvijas Grid projekta ietvaros ar dažādām metodēm veiktie Grid veiktspējas testi.



Saturs

1. SINTĒTISKIE TESTI	3
1.1. NETPIPE-TCP	3
1.2. DD	5
1.3. IOZONE	8
2. PAKALPOJUMU VEIKTSPĒJA	10
2.1. APACHE + PHP VEIKTSPĒJA	10
2.2. MYSQL VEIKTSPĒJA	11
2.3. SECINĀJUMI PAR XEN VEIKTSPĒJU	13
3. NOBEIGUMS	14

1. SINTĒTISKIE TESTI

Latvijas Grid projekta ietvaros tika veikta XEN tehnoloģiju veiktspējas testēšana. Šajā nodevumā ir atainota veiktspējas testēšanas metodoloģija un testu rezultāti. Par XEN tehnoloģiju vairāk ir rakstīts Latvijas Grid projekta nodevumā "Optimizētu XEN virtuālo mašīnu izmantošana".

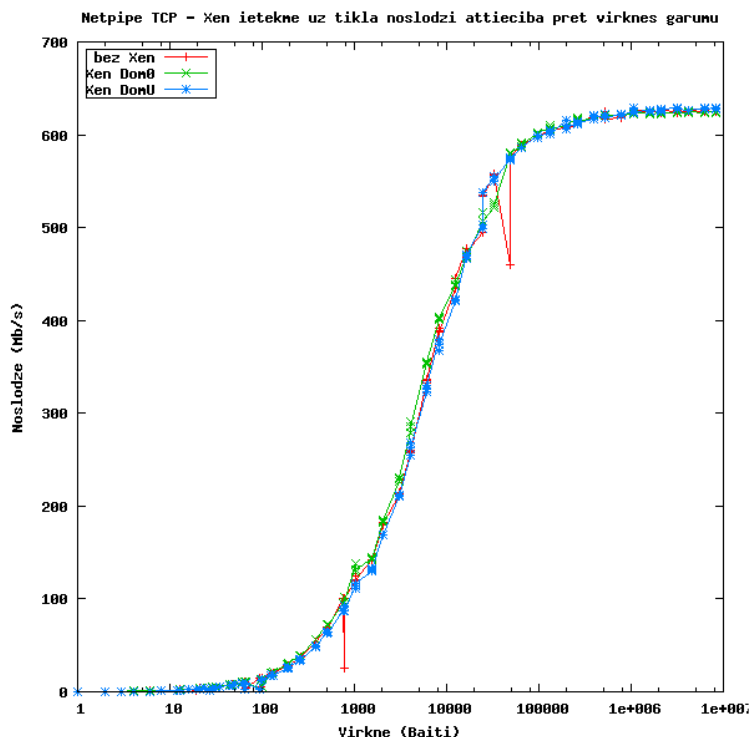
Pasaulē ir daudz rīku dažādu sistēmu aspektu veiktspējas testēšanai. Autors izvēlējās NetPIPE rīka TCP variantu, kas ir viens no visplašāk lietotajiem tīkla veiktspējas novērtēšanas rīkiem. Disku I/O sistēmas veiktspējas pārbaudei autors izvēlējās Unix sistēmās populāro ierīču informācijas kopēšanas rīku dd, kas parādīs informācijas rakstīšanas ātrumu ierīcē vizuālākajā līmenī, kā arī populāro rīku iozone, kas pārbauda vairāk nekā 10 faila operāciju (lasīt, pārrakstīt, lasīt lēkājot u.c.) veiktspēju ar dažāda izmēra blokiem.

1.1. NETPIPE-TCP

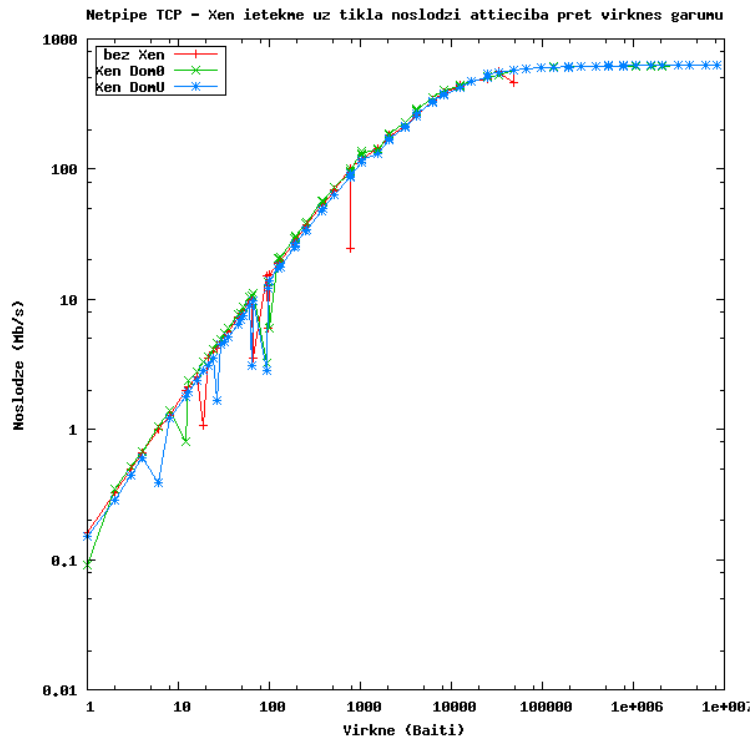
Autors izmantoja NetPIPE-TCP testus starp diviem jaudīgiem datoriem ar 1 Gb/s intrefeisiem, tos tieši savienojot ar vienu tīkla kabeli. Visos testos iniciators bija viens dators ar Ubuntu Linux Desktop operētājsistēmu, bet atbildētājs otrajā datorā bija Ubuntu Server 8.04 operētājsistēma palaista 3 dažādos veidos:

- ar noklusēto kodolu, neizmantojot Xen, un 4GB operatīvo atmiņu;
- ar Xen 3.2 bāzes mašīnā (Dom0) un 4GB operatīvo atmiņu;
- ar Xen 3.2 virtuālajā mašīnā (DomU) un 256MB operatīvo atmiņu.

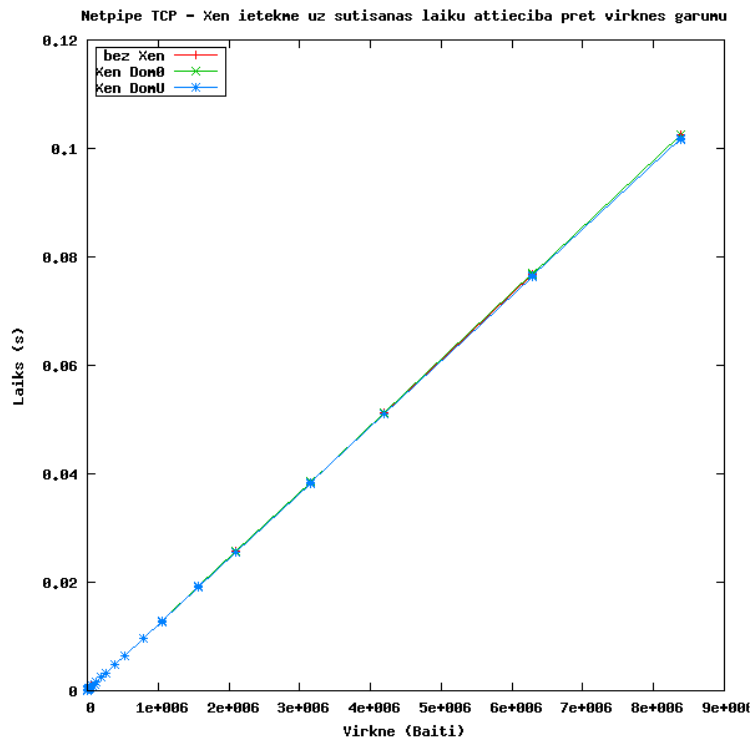
Turpmāk redzami salīdzinājuma attēli gan noslodzei (MB/s), gan dažādu virkņu garumu sūtīšanā patērētajiem laikiem. Attēli veidoti ar GNUplot programmatūru.



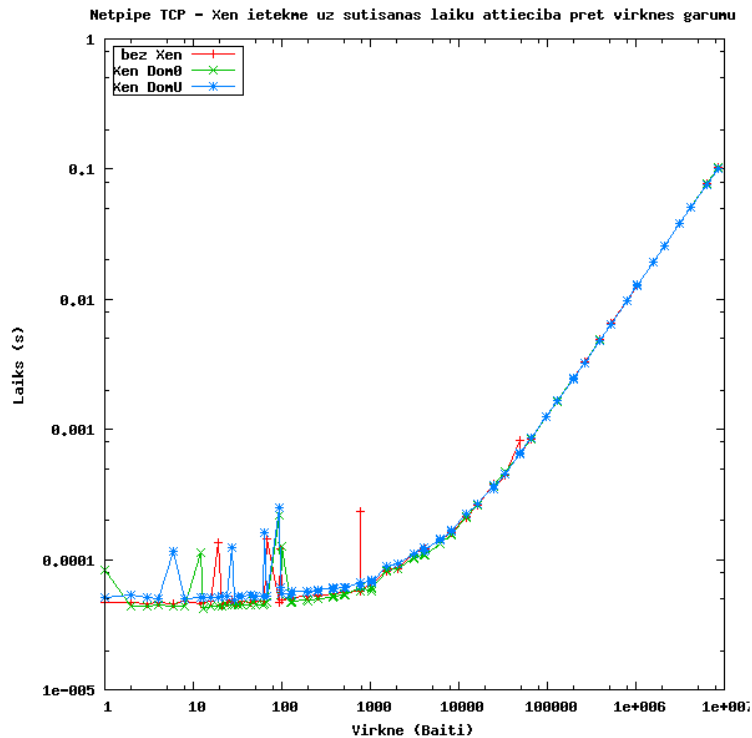
Attēls 1: TCP noslodze pret virknes garumu (log x)



Attēls 2: TCP noslodze pret virknes garumu (log x,y)



Attēls 3: TCP laiks pret virknes garumu



Attēls 4: TCP laiks pret virknes garumu (log x,y)

Attēlos redzamās svārstības visdrīzāk ir izkrikušas paketes rezultāts un uzskatāmas par izņēmumiem. Tendence ir acīmredzama: Xen izmantošana praktiski neietekmē tīkla veiktspēju. Citi secinājumi:

- Mazās virknēs DomU pievieno ~7-8 ms (mikrosekundes) lielu aizturi salīdzinājumā ar Dom0, bet ~3-4 ms lielu aizturi – salīdzinājumā ar konfigurāciju bez Xen. Tātad Dom0 nosūta nelielas paketes pat ātrāk par konfigurāciju bez Xen(!).
- Ap 100 KB virknes garumos sūtīšanas laiki kļūst ļoti līdzīgi un tādi saglabājas visos atlikušajos apjomos.
- DomU sasniedz lielāko tīkla noslodzi – 628-629 Mb/s (!).
- Apstiprinās viedoklis, ka parastos galda un klēpj datoros iebūvētie tīkla interfeisi ar nepielāgotu programmatūru spēj sasniegt 630 Mb/s tīkla noslodzi. Šādi skaitļi ir minēti vairākos avotos kā PCI kopnes reālā veiktspēja. Ar serveru klases PCI-E tīkla kartēm var sasniegt nominālo tīkla noslodzi.

1.2. DD

Ar šo rīku eksperimenti tika veikti diviem failu lielumiem: 1GB un 3GB. Šādi izskatās komanda 1GB faila veidošanai ar dd no 1MB lieliem blokiem no ierīces /dev/null, kas ģenerē 0 baitu straumi:

```
time dd if=/dev/zero bs=1024k of=/data1/dd.tmp count=1000
```

Komandai dd priekšā pielikta komanda time, kas pēc dd izpildes beigām parāda tās izpildes laiku ar precizitāti līdz milisekundai. Tāpat eksperimenti tikai veikti visām 3 interesējošām konfigurācijām:

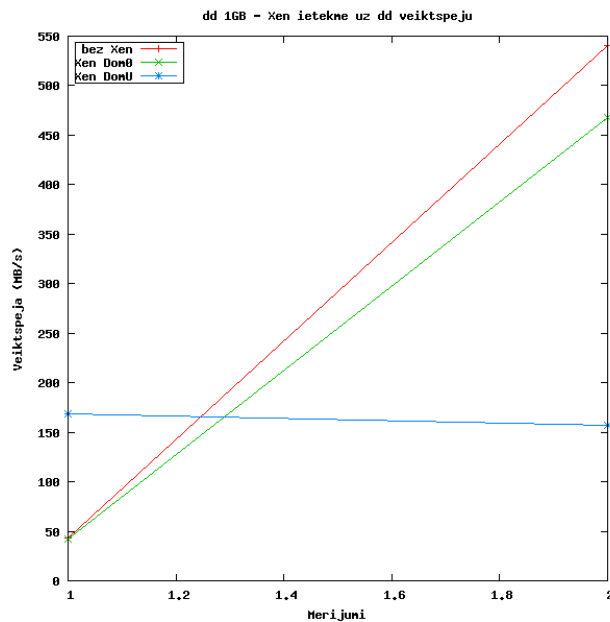
- noklusētajā kodolā (bez Xen) ar 4GB operatīvās atmiņas;
- Xen bāzes domēnā (Dom0) ar 4GB operatīvās atmiņas;

- Xen virtuālajā mašīnā (DomU) ar 2GB operatīvās atmiņas, izdalīta 4GB diska sadaļa.

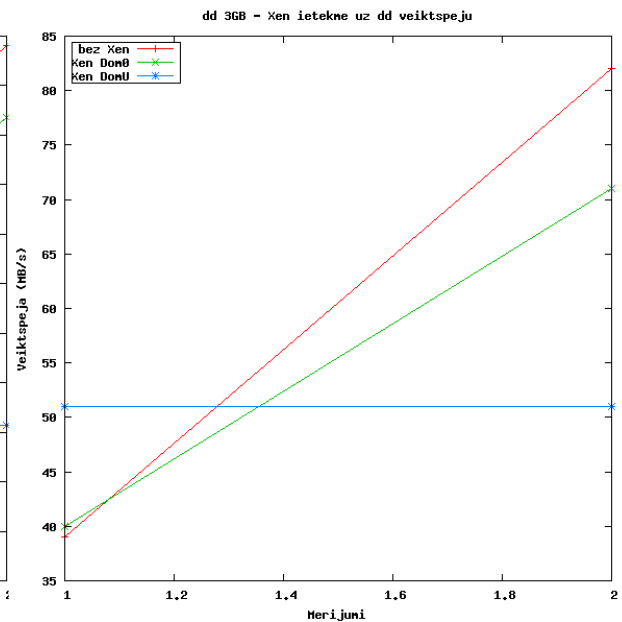
Lai novērotu arī kešatmiņas efektu uz dd veikspēju, katram eksperimentam tika veikti divi piegājieni pēc kārtas. Zemāk pievienotajā tabulā ir apkopoti testu rezultāti.

Faila izmērs	Mērījums p.k.	bez Xen		Xen Dom0		Xen DomU 2G	
		MB/s	sek.	MB/s	sek.	MB/s	sek.
1GB	1. mērījums	43,0	28,97	42,7	29,04	169	6,25
1GB	2. mērījums	541	5,58	468	3,10	157	6,89
3GB	1. mērījums	39,0	80,83	40,1	78,65	51,6	61,18
3GB	2. mērījums	82,6	39,71	71,3	44,68	51,8	76,42

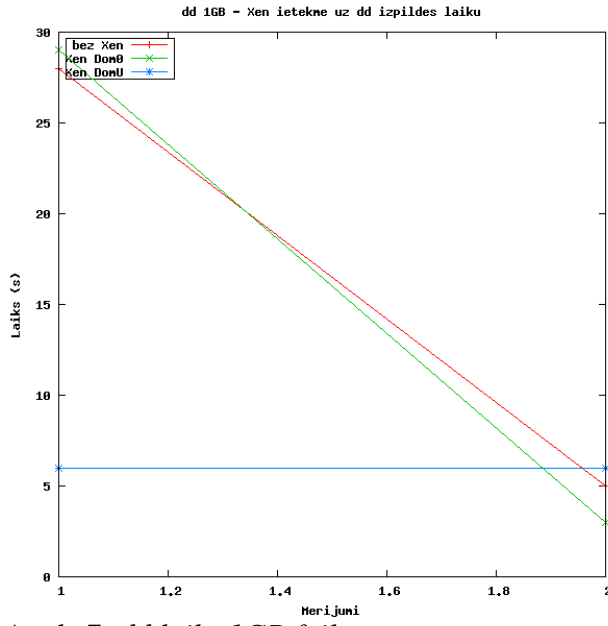
Rezultāti uzskatāmi redzami šādos attēlos:



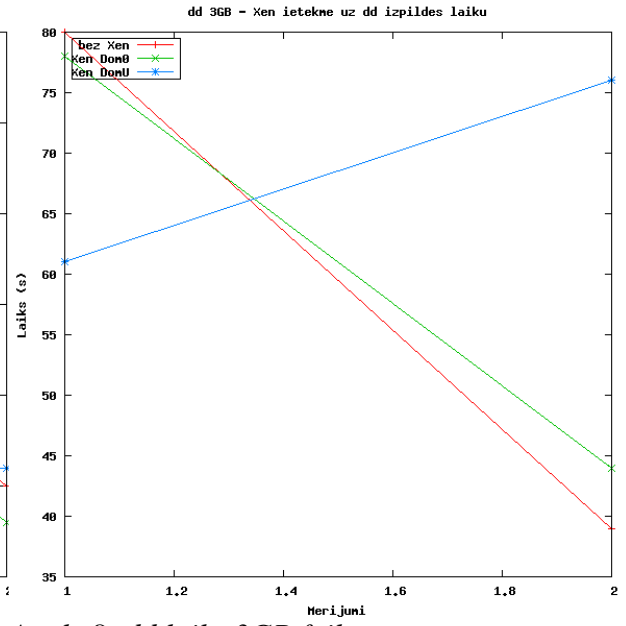
Attēls 5: dd veikspēja 1GB failā



Attēls 6: dd veikspēja 3GB failā



Attēls 7: dd laiks 1GB failā



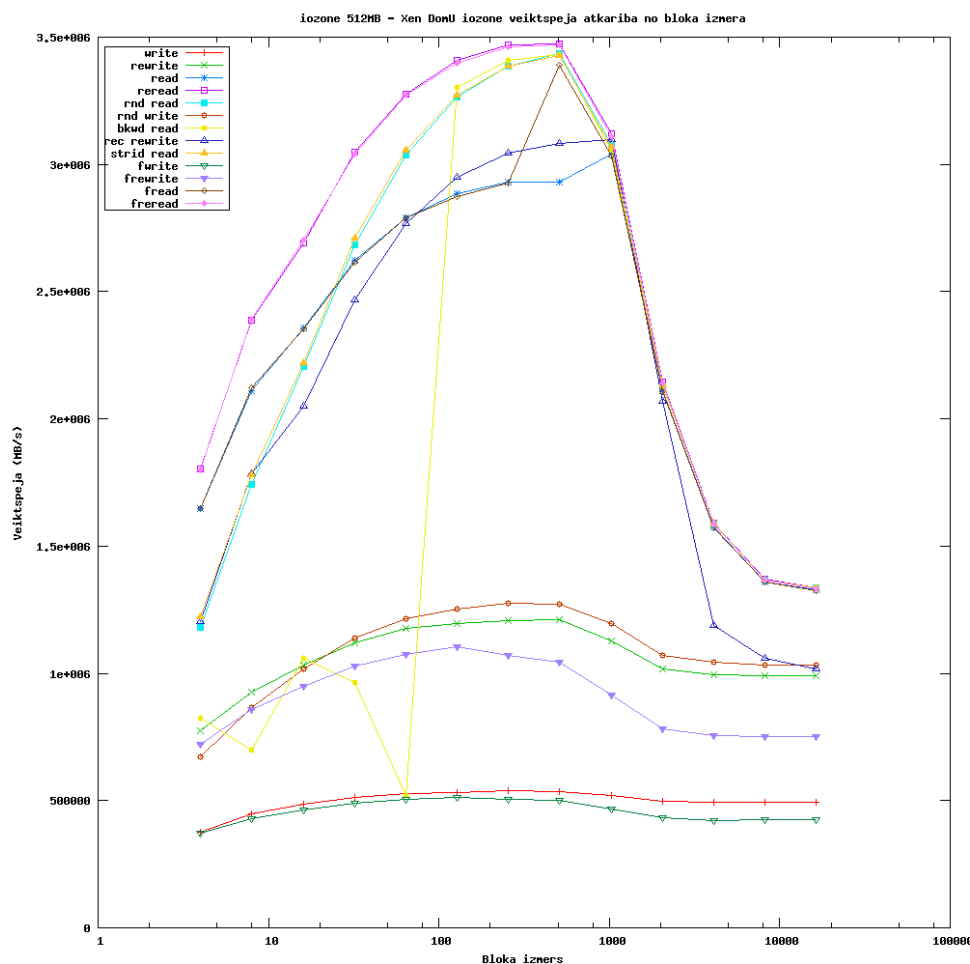
Attēls 8: dd laiks 3GB failā

Var izdarīt šādus novērojumus:

- Kešatmiņai ir milzīgs iespaids uz otro piegājienu bāzes mašīnā un datorā bez Xen. Šīm divām konfigurācijām 1GB failā veikspēja pieaug ap 10 reizes, bet 3GB failā – ap 2 reizes. Virtuālā mašīna tādu pieaugumu neuzrāda, abos piegājienos uzrāda līdzīgus rezultātus.
- Toties virtuālā mašīna jau ar pirmo piegājienu uzrāda lielāku veikspēju, kā pārējās konfigurācijas.
- Neviens mērījums neuzrāda veikspējas zudumus. Pat vājākie veikspējas rezultāti ir vismaz līdzvērtīgi maksimālai cietā diska veikspējai – 50MB/s.
- Secinājums: failu ģenerēšana ar dd ātrāk izpildās virtuālajā mašīnā; bet failu pārrakstīšana ar dd ātrāk izpildās Xen bāzes mašīnā un datorā bez Xen, pie tam līdzīgos laikos.

1.3. IOZONE

Tā kā šis rīks veic 14 dažādus faila operāciju testus, autors šeit tekstā neapkopos visus testu rezultātus. Šim eksperimentam atbilstošu diagrammu var redzēt attēlā:



Attēls 9: iozone veikspēja Xen virtuālajā mašīnā (log x)

Šis eksperiments veikts Xen virtuālajā mašīnā. Testa faila izmērs ir 512MB, bloku izmēri mainās no 4kB līdz 16MB. Tāpat kā iepriekšējos eksperimentos var novērot to, ka I/O sistēmas veikspēja Xen virtuālajā mašīnā sliktākajos gadījumos ir līdzīga fiziskā diska veikspējai, pārējos gadījumos – stipri augstāka.

Salīdzinājumu starp konfigurācijām var redzēt sekojošā tabulā. Tajā dots uz katru eksperimentu patērētais laiks. Tas parāda kopējo ātrumu uz var kalpot par apkopojošu salīdzinājumu. Salīdzināto konfigurāciju parametri ir tādi paši, kā iepriekš dd testā.

Faila izmērs	bez Xen, laiks	Xen Dom0, laiks	Xen DomU, laiks
64MB	2m2.639s	2m2.582s	2m3.137s
512MB	24m25.597s	24m23.978s	25m27.486s

Kā redzams, eksperimentu laiki ir ļoti līdzīgi. Xen virtuālā mašīna atpaliek no pārējām konfigurācijām, bet atpalcība ir niecīga un var netikt ņemta vērā.

2. PAKALPOJUMU VEIKTSPĒJA

2.1. APACHE + PHP VEIKTSPĒJA

Gan bāzes domēnā, gan testa domēnā autors instalēja Apache 2.0, MySQL, un divas Tīmekļa programmas: Mediawiki un Wordpress. Tās abas var iegūt no Ubuntu repozitorijiem, un tika instalētas ar standarta komandām:

```
apt-get install apache2
apt-get install wordpress mediawiki
```

Abu programmu sākotnējā noskaņošana ir vienkārša un labi aprakstīta to mājas lapās. Testēšanai autors izmantoja šādus objektus:

- Apache noklusētā lapa – <http://serveris/index.html>
- Apache servera statusa lapa – <http://serveris/server-status>
- Wordpress galvenā lapa – <http://serveris/wp/index.php>
- Mediawiki galvenā lapa – http://serveris/mediawiki/index.php/Main_Page

Testa domēnam tika izdalīti divi procesori, 512MB operatīvās atmiņas, 1GB virtuālais disks, 256MB virtuālās atmiņas.

Pieprasījumus autors ģenerēja ar programmu ApacheBench [ab] no cita jaudīga datora tajā pašā lokālajā tīklā. Fiziskā konfigurācija bija ļoti līdzīga iepriekš tīkla testos izmantotajai: abi datori pievienoti vienam 1Gb/s komutatoram ar tādas pašas joslas interfeisiem, nodrošinot īsu 1 Gb/s ceļu no klienta līdz serverim.

Ar ApacheBench komandu var pārbaudīt lapas atvēršanas veiktspēju:

```
ab -c 1 -n 500 http://serveris/wp/index.php
```

Parametrs -c norāda vienlaicīgo pieprasījumu daudzumu, parametrs -n – kopējo pieprasījumu daudzumu. Tīmekļa programmām tika veiktas trīs testu grupas: ar vienu, diviem, un ar desmit vienlaicīgiem pieprasījumiem. Statiskajai lapai un servera statusam tika testēti arī 50 vienlaicīgi pieprasījumi, pie tam kopējais pieprasījumu daudzums bija 5000. Rezultāti apkopoti pievienotajā tabulā, kurā katrā rūtiņā ir abu lapu atsaukšanās daudzumi sekundē, pirmais – Wordpress, otrais – Mediawiki.

Tests	bez Xen	Xen Dom0	Xen DomU 512	Xen DomU 3G
index.html: 1	2572,35	2612,82	2280,69	2351,06
index.html: 2	4073,78	4082,97	3716,94	3951,66
index.html: 10	13675,59	8353,79	5909,61	6285,45
index.html: 50	14056,55	8708,63	5614,78	6257,74
status: 1	610,95	694,90	658,20	669,10
status: 2	1112,03	1214,11	1081,06	1136,10
status: 10	2088,72	2172,32	1950,18	2001,93
status: 50	1977,68	2145,30	2060,34	2052,27
Wordpress: 1	9,79	9,68	9,98	9,71
Wordpress: 2	18,92	17,93	18,80	18,68

Tests	bez Xen	Xen Dom0	Xen DomU 512	Xen DomU 3G
Wordpress: 10	19,22	19,04	19,07	18,60
Mediawiki: 1	6,65	6,63	6,63	6,43
Mediawiki: 2	12,79	11,89	12,02	11,60
Mediawiki: 10	13,28	13,05	12,91	12,53

Autors secina, ka:

- Komplicētas Tīmekļa PHP programmas, kas lapas ģenerēšanai izpilda vairākus PHP skriptus ar desmitiem SQL pieprasījumu uz MySQL serveri, strādā gandrīz vienlīdz ātri visās Xen konfigurācijās, kā arī bez Xen.
- Arī Apache paša ģenerētā statusa lapa atbild vienlīdz ātri visās konfigurācijās.
- Būtiska atšķirība ir tikai 10 un vairāk vienlaicīgiem pieprasījumiem pēc statiskās index.html lapas. Dators bez Xen sasniedz 14 tūkstošus lapu sekundē, kamēr Xen konfigurācijas spēj apstrādāt tikai 6-8 tūkstošus.
- Mašīna ar 512 MB atmiņu strādā nedaudz ātrāk par Xen bāzes sistēmu un mašīnu ar 3GB atmiņas.

2.2. MYSQL VEIKTSPĒJA

MySQL izstrādātāji ir publicējuši dokumentu par MySQL veiktspējas novērtēšanu, kurā arī izmanto atklātā pirmkoda rīku SysBench. Par SysBench lietošanu MySQL veiktspējas un mērogojamības novērtēšanai ir aprakstīts arī citās vietnēs.

Autors lietoja standarta bināro MySQL pakotni i386 arhitektūrai, neko neoptimizējot un īpaši nekonfigurējot, jo uzdevums ir salīdzināt dažādās virtualizācijas konfigurācijas.

Pirmos testus autors veica ar nesavietojamu TLS bibliotēku. Tas palēnināja Xen konfigurāciju lasīšanas testus par apmēram 10%. Vēlāk gan uz bāzes, gan viesu mašīnas aizliedza ar Xen nesavietojamo TLS bibliotēku ar zemāk norādīto komandu, kas deva tālāk aprakstītos rezultātus.

```
mv /lib/tls /lib/tls.disabled
```

MySQL veiktspējas pārbaudīšanai autors lietoja SysBench rīku, kuru ļoti bieži izmanto šiem mērķiem. Autors izveidoja jaunu mašīnu ar diviem virtuālajiem procesoriem, 1GB lielu virtuālo failsistēmu un 256MB operatīvās atmiņas. SysBench pieslēdzās MySQL serverim lokāli. Abos domēnos autors ieinstalēja MySQL serveri ar šādu komandu:

```
apt-get install mysql-server
```

Tad izveidoja testiem nepieciešamo datubāzi ar nosaukumu sbtest:

```
%mysql -u root -h localhost
mysql> create database sbtest;
mysql> exit;
```

Tā kā SysBench nav iekļauts Ubuntu repozitorijā, autors lejupielādēja tā pirmkodu no SysBench lapas failā sysbench-0.4.8.tar.gz. Pirms kompilēt to, nepieciešama kompilēšanas programmatūra un MySQL izstrādes .h faili, kurus pieinstalē ar sekojošām komandām:

```
apt-get install build-essential
apt-get install libmysqlclient15-dev
```

Tad var ķerties pie SysBench atpakošanas, kompilēšanas:

```
tar xzf sysbench-0.4.8.tar.gz
```

```
cd sysbench-0.4.8
./configure
make
make install
```

Tālāk autors sagatavoja testu datubāzi, aizpildot to ar 1 miljonu ierakstu:

```
sysbench --test=oltp --mysql-user=root --oltp-table-size=1000000 prepare
```

Tik liela datubāze aizņem apmēram 256 MB diska vietas. Lai izpildītu testu ar vienu pavedienu, lieto šādas divas komandas (pirmā – ar rakstīšanu 10 tūkst. pieprasījumiem, otrā – tikai lasīšanu 100 tūkst. pieprasījumiem, jo tie izpildās ātrāk):

```
sysbench --num-threads=1 --max-requests=10000 --test=oltp --mysql-user=root --oltp-table-size=1000000 run
```

```
sysbench --num-threads=1 --max-requests=100000 --test=oltp --mysql-user=root --oltp-table-size=1000000 --oltp-read-only run
```

Šādus testus autors veica ar dažādiem paralēlo pavedienu daudzumiem uz viena datoru, kas palaists 4 dažādos veidos: bez Xen ar 4GB atmiņas, Xen bāzes mašīnā (Dom0) ar 4GB atmiņas, Xen mašīnā (DomU) ar 256MB atmiņas, un Xen mašīnā (DomU) ar 3GB atmiņas. 256MB mašīnā nepietiek brīvās atmiņas, lai visu datubāzes informāciju no diska turētu kešatmiņā. Tādēļ autors veica eksperimentu arī ar 3GB atmiņu, lai varētu pilnībā izmantot kešošanu. Rezultātu tabulas rezultātu rūtiņās rakstīti divi skaitļi: transakcijas sekundē ar rakstīšanu, transakcijas sekundē tikai lasīšanai.

Pavedieni	bez Xen	Xen Dom0	Xen DomU 256	Xen DomU 3G
1	109, 414	109, 397	105, 410	107, 398
4	101, 913	102, 754	101, 785	101, 756
16	101, 833	100, 611	105, 622	105, 599

Pirmkārt jāpiemin, ka cietņa aktivitātes indikators ļoti aktīvi mirgoja rakstīšanas testos, bet ļoti minimāli lasīšanas testos. Tas piedod šiem testiem papildus nozīmi. Rakstīšanas tests parāda arī disku I/O sistēmas darbību, bet lasīšanas tests parāda informācijas apstrādes ātrumu operatīvajā atmiņā.

Var secināt, ka datubāzes rakstīšanas operācijas uzrāda ļoti līdzīgu veikspēju visās konfigurācijās. Tātad arī darbs ar disku uzrāda līdzīgu veikspēju, pat neskatoties uz to, ka virtuālā failsistēma glabājas failā, nevis izdalītā reālā diska sadaļā. Lasīšana viena pavediena ietvaros parāda, ka visas Xen konfigurācijas strādā par pāris procentiem lēnāk par datoru bez Xen, pie tam gandrīz nav starpības starp "Dom0" un "DomU-3G". Bet mazā 256MB mašīna uzrāda līdzīgu rezultātu datoram bez Xen. Vispār visos lasīšanas testos mazā mašīna ir nedaudz ātrāka.

Toties lasīšanas veikspējas starpība pieaug, ja darbina vienlaicīgi vairākus pavedienus. Vislielāko veikspēju uzrāda 4 pavedieni, kuru gadījumā Xen konfigurācijas ir par apmēram 17% lēnākas par datoru bez Xen. 16 pavedienu gadījumā starpība jau ir virs 20%, kas acīmredzami norāda uz Xen grūtībām apkalpot daudzus paralēlus aktīvi ar operatīvo atmiņu strādājošus procesus.

Autors izdara šādus secinājumus MySQL serveru izmantošanai Xen virtuālajās mašīnās:

- Diska I/O veikspēja reālos uzdevumos nesamazinās.

- Nav vērts mašīnām dot pārmērīgi lielus operatīvās atmiņas apjomus, tās lieliski strādā arī ar pieticīgu atmiņas apjomu. Tās pat labāk tiek galā ar tām piemērotu atmiņas apjomu, nekā pārspīlēti lielu.
- Ja paralēlu aktīvu procesu skaits ir krietni lielāks par procesoru/kodolu daudzumu, veiktspējas zudumiem ir liela varbūtība.
- Vismaz MySQL+SysBench testi parāda, ka MySQL lietošana ar daudziem vienlaicīgiem pieprasījumiem Xen domēnā zaudē līdz 20% veiktspējas.
- Kopumā – Xen ir veiksmīgi lietojams nelielas un vidējas slodzes MySQL serveros.

2.3. SECINĀJUMI PAR XEN VEIKTSPĒJU

Xen virtuālās mašīnas ir izmantojamas lielākajai daļai serveru un skaitļošanas uzdevumu. Tīkla veiktspēja ir nevainojama, disku I/O sistēma strādā labi. Sarežģītas Tīmekļa programmas ar datubāzes atbalstu Xen mašīnās strādā īpaši labi. Arī pārējie lietojumi strādā stabili un ar minimālu veiktspējas zudumu, izņemot sekojošus gadījumus, kad nav ieteicams lietot Xen:

- Apache serveris nelieliem statistiskiem objektiem, piemēram, HTML lapām vai nelielām bildēm. Testi rāda, ka Xen mašīnās šāda uzdevuma veiktspēja ievērojami krītas.
- Arī augstas slodzes MySQL serveri ar lielu daudzumu vienlaicīgu transakciju ieteicams darbināt uz datora bez Xen.

Autors izvirza hipotēzi, ka ievērojami veiktspējas zudumi ir saistīti ar biežu (desmitiem tūkstošu reižu sekundē) kontekstu pārslēgšanu starp uzdevumiem un to procesiem. Acīmredzot Xen papildu slānis šo procesu kavē. Bet reālā dzīvē tādi scenāriji, kuros tiešām nepieciešama tik aktīva pārslēgšanās, ir retums. Tiem ieteicams rezervēt izdalītus fiziskus datorus ar ātru operatīvo atmiņu un tās kontrolieri, kā arī ātriem cietajiem diskkiem.

Tā kā lielākai daļai uzdevumu Xen virtuālās mašīnas praktiski nemazina veiktspēju, autors iesaka to lietot visu skaitļošanas uzdevumu iekļaušanai virtualizētās operētājsistēmās.

3. NOBEIGUMS

Latvijas Grid projekta ietvaros veiktie pētījumi tika publicēti maģistra darbā L.Trukšāns "Jaunas paaudzes skaitļošanas infrastruktūras principi", vadītājs Dr.dat. Guntis Bārzdiņš. Šajā nodevumā ir izmantoti maģistra darba materiāli.