Testování modelu Object-based Storage Device pro OpenAFS

Michal Švamberg
Luboš Kejzlar

Západočeská univerzita v Plzni
Centrum informatizace a výpočetní techniky
e-mail: [svamberg, kejzlar]@civ.zcu.cz

2. června 2010

Obsah

Technická zpráva je výstupem projektu Fondu rozvoje CESNET, z.s.p.o vedeným pod číslem 293/2009. Dokument popisuje testovací prostředí, metodiku jednotlivých testů a získané výsledky.

1 Cíle testování

Cílem testování bylo ověřit funkčnost všech komponent rozšíření Rx-OSD, jejich stabilitu a vhodnost nasazení v produkčním prostředí univerzity.

2 Úvod k Rx-OSD

Rx-OSD je následník MR-AFS, který byl vyvinut na Pittsburgh Supercomupting Center. Nyní je vyvíjen Hartmutem Reuterem z Rechenzentrum Garching (RZG). Základem je přidání nové infrastruktury, která soubory ukládá na servery OpenAFS. Oproti klasickému fileserveru jsou soubory uloženy na OSD serverech. Tato vlastnost umožňuje mít více RW kopii souboru, ukládat soubory do hierarchických diskových systému (HSM) nebo automaticky migrovat soubory z disků na pásky. Soubory mohou být interně rozděleny a rozloženy přes více OSD serverů, pak hovoříme o objektech než o souborech. Objekty mohou mít uvnitř OSD více kopií.[3]

3 Popis testovací prostředí

Testovací prostředí se skládalo ze serverové a klientské části. Technické vybavení serverové části byla pořízeno z prostředků projektu FR CESNET z.s.p.o číslo 293/2009. Klientskou část tvořila vysoce výkonná stanice a cca. 30 standardních učebnových PC v různých konfiguracích.

3.1 Návrh infrastruktury

Vzhledem k omezeným prostředkům byla testovací infrastruktura založena na využití virtualizačních technologií. Návrh byl proveden tak, aby se jednotlivé virtuální stroje minimálně ovlivňovali a nedocházelo ke vzniku „úzkých míst“.

Nejvýznamnějším faktorem je výkon diskového subsystému. Proto pro každý DomU[1] byl přiřazen jeden fyzický HDD a do systému byly zakoupeny kapacitně menší disky s maximálním možným výkonem. Každý disk byl přiřazen do samostatné volume group (VG) v LVM spravované z Dom0. VG byla rozdělena na kořenový systém (20 GB), swapový oddíl (2 GB) a oddíl pro AFS fileserver tzv. /vicepa (50 GB). Zbylý prostor tvořil rezervu pro případné další testy. Na discích sda a sdb bylo navíc rezervováno cca. 45 GB pro vytvoření

[1]Neprivilegovaný virtuální stroj. Opakem je Dom0, který má oprávnění nastavovat přístup k hardware ostatním virtuálním strojům.

mirroru přes MD-RAID, který byl dedikován Dom0 systému. Toto rozdělení disků nepředstavuje výkonnostní problém, protože Dom0 se používá pro správu virtuálních strojů a neúčastní se testování.

Systém Dom0 měl přiděleno 4 GB RAM aby bylo dosaženo minimalizace diskových operací díky využití cache. Opačný přístup byl zvolen pro konfiguraci DomU systémů, které měly přidělen 1 GB RAM s cílem minimalizovat vliv interních cachovacích mechanizmů na výsledky testů.

Použitý procesor E5504 má čtyři fyzická jádra (nevyužívá hyperthreading), každé jádro bylo konfiguračně pevně svázáno s jedním DomU. Řídící Dom0 sdílel všechny dostupné procesory, čímž bylo dosaženo snížení zátěže jednotlivých procesorů a jejího rovnoměrného rozložení.

Testovací servery měly k dispozici pouze dvě 1 Gb síťové karty. DomU byly tedy přiděleny síťovým kartám po dvojici přes bridge. Dom0 jsme připojili přímo k bridge na eth0. Zde bylo potřeba zajistit, aby propustnost připojení nebyla nižší než propustnost ze dvou DomU současně.

3.2 Serverová infrastruktura

3.2.1 Technické vybavení

Z grantu byly pořízeny dva standardně vybavené fyzické stroje (viz tab. 1). Každý z nich byl rozdělen virtualizační technologií Xen[2] na čtyři samostatné DomU servery, které poskytovaly všechny nezbytné služby infrastruktury.

Rozdělení a označení serverů:

chryso1, chryso2 jsou fyzické servery. Zároveň realizují řídící virtuální stroje Dom0, určené pouze pro management virtuálních strojů DomU.

chryso1-1, chryso1-2, chryso1-3, chryso1-4 jsou virtuální stroje DomU na serveru chryso1.

chryso2-1, chryso2-2, chryso2-3, chryso2-4 jsou virtuální stroje DomU na serveru chryso2.

Celková konfigurace virtualizované infrastruktury je znázorněna na obr. 1, síťové zapojení ilustruje obr. 2.

3.2.2 Operační systém

Serverová část byla postavena na architektuře x86-64 (amd64) na operačním systému Debian GNU/Linux (Lenny). Mimo standardní instalaci byla použita vlastní kompilace jádra 2.6.31.8 s rozšířením o podporu Dom0 projektu Xen verze 3.4 a OpenAFS server s podporou Rx-OSD ve verzi 1.4.12.

[2] `http://www.xen.org`

Konfigurace fyzického stroje

RAM	8 GB, 1066 MHz
CPU	1× Intel Xeon E5504 (2.0 GHz, 4 M Cache, 4.86 GT/s QPI)
HDD	4× 300 GB SAS 15 k 3.5“
NET	2× 1 Gb/s Broadcom BCM5716

Konfigurace virtuálního stroje – Dom0

RAM	4 GB
CPU	4× jádro, každé sdílené s jedním DomU
HDD	2× dedikovaná partition ze 2 různých HDD do mirroru softwarového RAIDu
NET	1× 1 Gb/s sdílená se dvěma DomU

Konfigurace virtuálního stroje – DomU

RAM	1 GB
CPU	1× dedikované jádro
HDD	1× dedikovaná partition, HDD není sdílen s jiným DomU, LVM z Dom0
NET	1× 1 Gb/s sdílená s druhým DomU

Tabulka 1: Základní hardwarová konfigurace serverů.

3.3 Klientská část

Konfigurace jednotlivých klientských stanic je přehledně uvedena v tab. 2.

3.3.1 Operační systém

Klientská část byla postavena na OS Debian GNU/Linux (architektura x86 i amd64) s jádrem 2.6.30 a OpenAFS klientem ve verzi 1.4.11 s rozšíření Rx-OSD.

3.4 Konfigurace OpenAFS

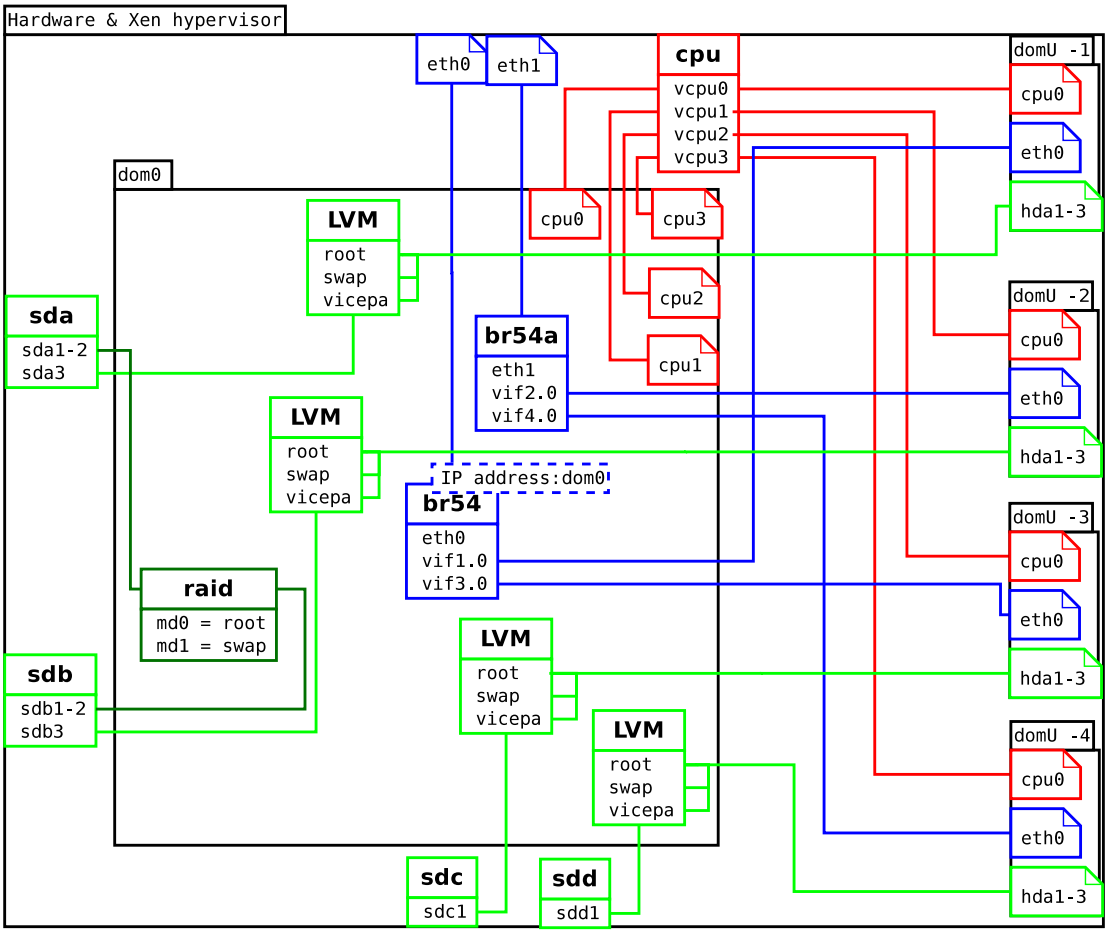
Veškeré testování probíhalo v samostatné AFS buňce civ.zcu.cz oddělené od provozní infrastruktury.

Na všech serverech chryso1-x a chryso2-x byla nainstalována služba rxosd. Na server chryso1-1 byly navíc nainstalovány služby volserver, osddbserver, vlserver, ptserver a fileserver. Tento server zároveň sloužil jako autentizační kerberos server.

Pro testování byla vytvořena sada volumů s různými politikami:

stripe0 bez politiky. Složí k testování přímého přístupu na fileserver bez použití rozšíření Rx-OSD.

stripe1,2,4,8 jeden, dva, čtyři nebo osm stripe na data spravovaná rxosd servery.

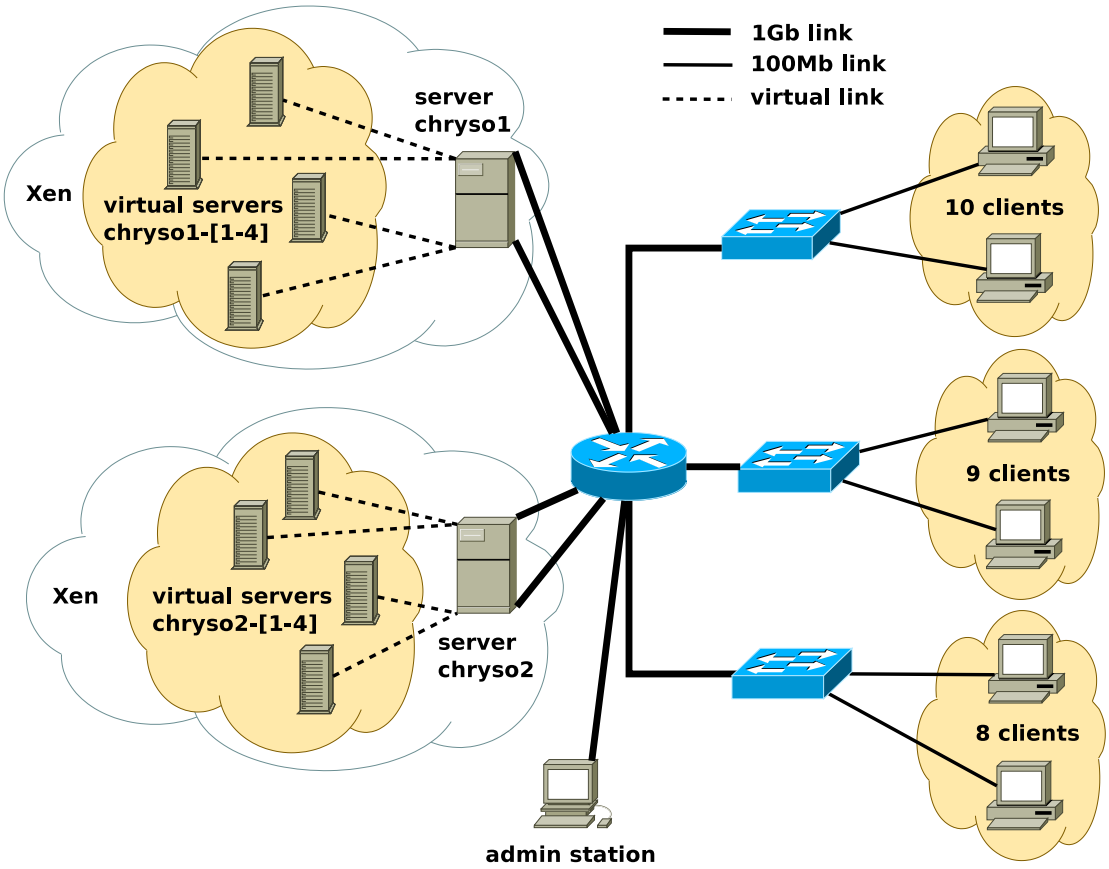


Obrázek 1: Konfigurace virtualizace.

Velikost stripe[3] byla vždy 12 ($2^{12} = 4096\,\mathrm{B}$), použitý počet stripe je mocninou 2 a je implementačně omezen na maximálně 8. U variant stripe1-8 slouží fileserver pouze k uložení metadat o souborech, vlastní správa dat je ponechána na rxosd serverech, na které rozkládá zátěž přímo klient. Pokud klient nepodporuje Rx-OSD rozšíření, pak tuto funkcionalitu rozložení dat za něj zprostředkovává fileserver.

Z výše uvedených údajů je patrné, že při použití stripe0 se bude s daty nakládat „klasickým způsobem“, zatížen bude pouze server chryso1-1, na kterém běží fileserver. Naopak u RxOSD variant stripe1-8 se bude zátěž rozkládat mezi všechny dostupné rxosd servery.

[3]Je počet stripů (kusů) ze kterých se skládá soubor. S 2 stripy a velikostí stripu (stripesize) 12 bude první 4 kB vložen do stripu 0, další 4 kB do stripu 1 a třetí opět do stripu číslo 0, a tak dále. Každý strip je objekt na různém OSD.



Obrázek 2: Síťové zapojení testovacího prostředí.

4 Popis testování

Samotné testování bylo velmi časově náročné[4] a vzhledem k provoznímu využívání učeben mohlo probíhat pouze v nočních hodinách.

Pro generování V/V datového toku požadovaných vlastností byl použit program iozone, který byl na klientech spouštěn příkazem parallel-ssh:

```
time parallel-ssh -h nodes.txt -p 100 -t 86400 -o result 'iozone_command'
```

Položka `iozone_command` vypadala následovně:

```
iozone -s 1G -r 256k -c -t 1 -F ‘tempfile -d stripe2‘ -i 0 -i 1 -i 2
```

přičemž jednotlivé parametry mají následující význam:

`-s 1GB` velikost testovacího souboru 1 GB,

[4]Některé testy trvaly až několik hodin.

Učebna UI505b – 9 strojů

RAM	1 GB
CPU	1× Intel Pentium 4, 3.40 GHz, jedno jádro
NET	1*times* 100 Mb/s do switche s 1Gb/s uplinkem
ARCH	32-bit

Učebna UI505 – 8 strojů

RAM	2 GB
CPU	1× AMD Athlon 64 X2 Dual Core 4200+, 1 GHz, dvě jádra
NET	1× 100 Mb/s do switche s 1 Gb/s uplinkem
ARCH	64-bit

Učebna UI312 - 10 strojů

RAM	768 MB
CPU	1× Intel Pentium 4 CPU, 2.60 GHz, jedno jádro
NET	1× 100 Mb/s do switche s 1 Gb/s uplinkem
ARCH	32-bit

Tabulka 2: Hardwarová konfigurace klientů.

-r 256k velikost záznamu pro operace je 256 kB. Tato hodnota se v testech měnila.

-c do výsledného času bude započítán také čas potřebný pro funkci close(),

-t 1 počet vláken, které mají provádět operace souběžně,

-F filename název souboru, zde generovaný příkazem tempfile v adresáři stripe2,

-i 0 bude proveden test typu write/rewrite,

-i 1 následně bude proveden test typu read/reread,

-i 2 jako poslední provede iozone test typu random-read/write.

Každý test byl prováděn opakované a následně byl vyhodnocen:

Celkový čas z příkazu time. Je to čas potřebný pro dokončení celého testu, tj. všech operací na všech nodech.

iozone *statistika* z každého nodu, který byl do testu zahrnut. Znázorňuje rozptyl zpoždění daných operací mezi jednotlivými nody.

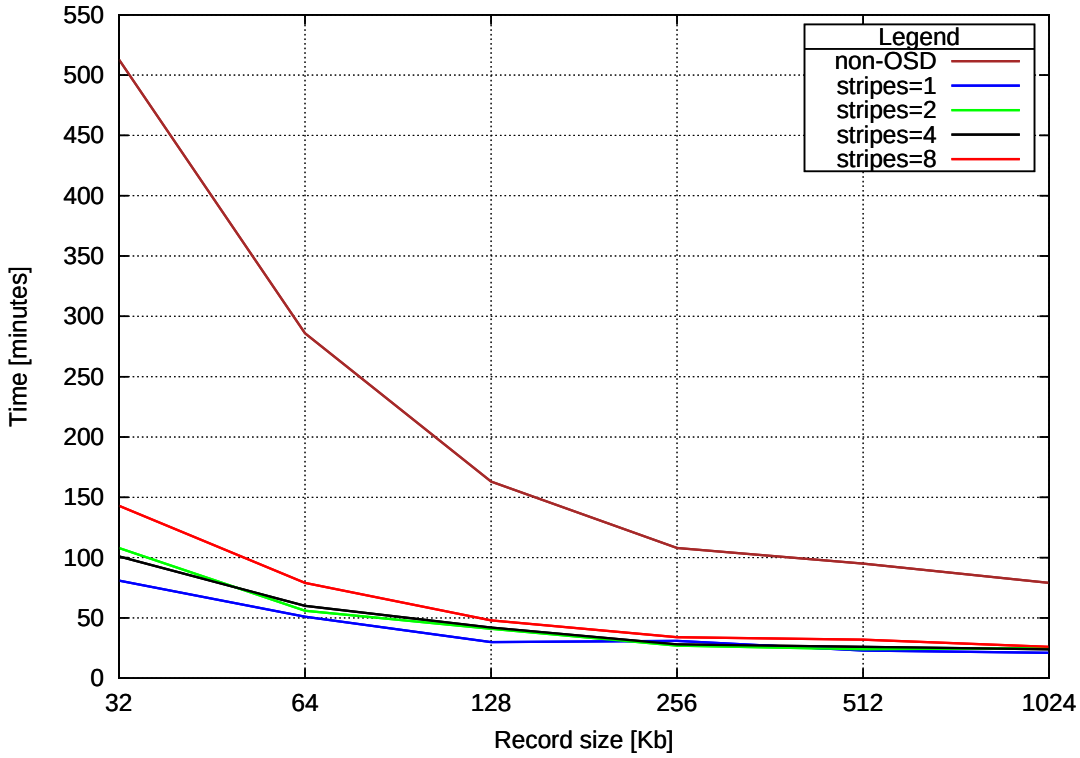
***Zátěž na* Dom0 chryso1 *a* chryso2** měřená programem dstat. Statistiky zatížení disků a síťového rozhraní měřené ve vteřinových intervalech.

5 Výsledky testů

Všechny níže uvedené výsledky jsou vztaženy vůči času, který byl potřebný pro dokončení celého testu, tj. všech zadaných úloh na všech nodech.

5.1 Vliv velikosti bloku

V tomto testu se měnila velikost bloku, s kterým iozone pracoval (parametr -r, record size). S menším blokem je potřeba více operací a tudíž více času. Ve výsledném grafu na obr. 3 je patrný rozdíl ve výkonnosti mezi Rx-OSD a klasickým OpenAFS.



Obrázek 3: Vliv změny velikosti bloku přes různé nastavení počtu stripů.

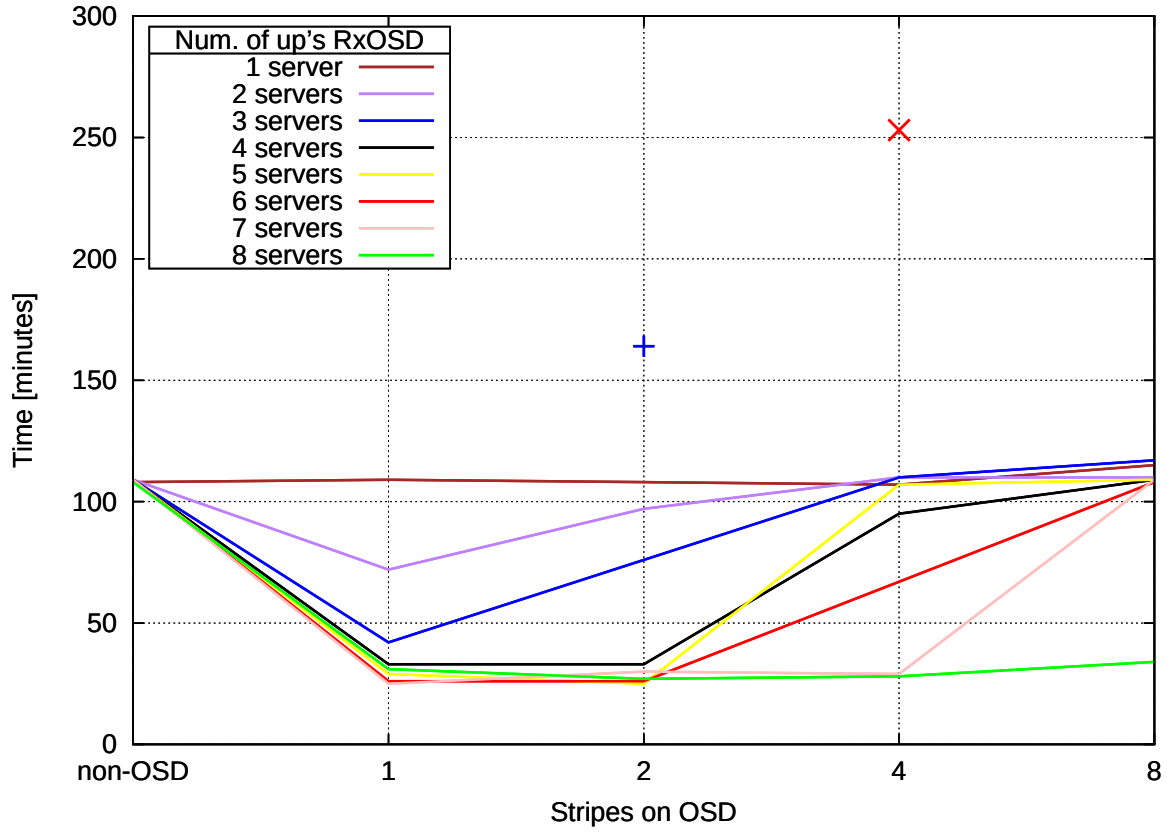
5.2 Vliv odpadnutí rxosd serveru

Tento test byl zařazen, aby bylo odzkoušeno chování systému v případě neočekávaného výpadku rxosd serveru.

Grafy na obr. 4 a 5 jsou vytvořeny nad stejnými daty. V průběhu zpracování a po konzultaci s vývojáři Rx-OSD byly 2 extrémní hodnoty aproximovány (originální hodnoty vyznačeny jako body). Důvodem byla chyba měření a nedostatečný časový odstup mezi

výpadky jednotlivých rxosd serverů, kde pak byl aplikován timeout na odezvu vypnutého serveru.

Z grafů vyplývá, že žádný případ (vyjma výše popsaných vyjímek) není výrazně horší než klasické OpenAFS. Rx-OSD vykazuje většinou lepší nebo nejhůře stejnou propustnost.

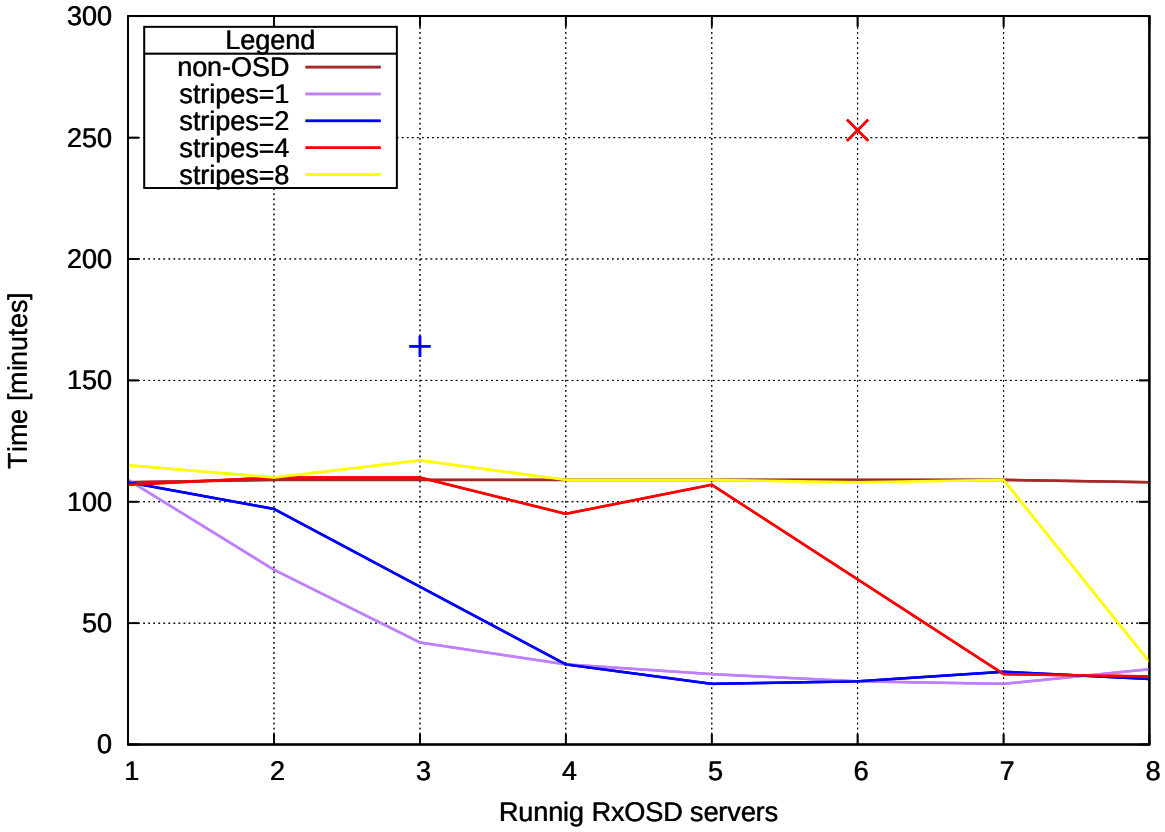


Obrázek 4: Vliv velikosti stripu na počet běžících serverů.

5.3 Vliv počtu klientů

V průběhu testu byl postupně zvyšován počet klientů od 1 do 27. Konfigurace Rx-OSD byla zvolena pro počet stripů 2 a v provozu bylo všech 8 rxosd serverů. Toto nastavení vyšlo z předchozích měření jako nejstabilnější a pravděpodobně nejvhodnější pro využití v produkčním prostředí univerzity.

Z obr. 6 je patrné, že úzké místo v propustnosti zpočátku tvoří klientské stanice a při použití standardního OpenAFS se brzy přesouvá na fileserver. Naproti tomu Rx-OSD vykazuje v testovaném pásmu prakticky konstantní propustnost.



Obrázek 5: Vliv počtu běžících serverů na velikosti stripu.

5.4 Zátěž disků a síťových rozhraní

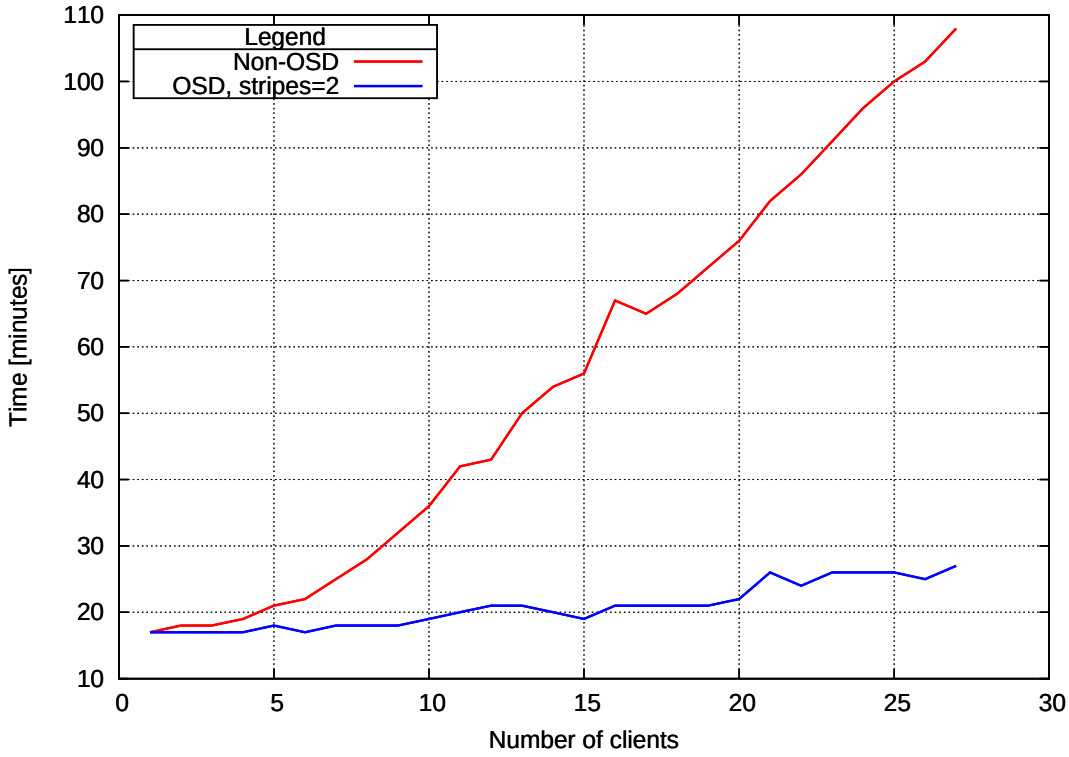
Na obr. 8 a 7 je znázorněn průběh zatížení disků a zátěže na síťových rozhraních měřené na Dom0. Data odpovídají testu při zapnutém Rx-OSD s 27 klienty a 8 rxosd servery.

Z grafů je patrné, že v dané konfiguraci nebyla propustnost síťového rozraní omezujícím faktorem. Naopak výkonnost diskového subsystému je významným faktorem ovlivňujícím výsledky testů.

6 Závěr

K výsledkům testů je třeba poznamenat, že virtualizovaná infrastruktura fileserverů vyžaduje určitou režii provozu. Lze tedy očekávat, že absolutní propustnost bude v produkčním prostředí vyšší. Avšak pro porovnání poměru Rx-OSD s klasickým souborovým serverem OpenAFS bylo navržené testovací prostředí zcela vyhovující.

Hlavním cílem projektu bylo seznámení se s technologií Rx-OSD a posouzení možného přínosu a vhodnosti nasazení v produkčním prostředí Západočeské univerzity v Plzni. Z vý-

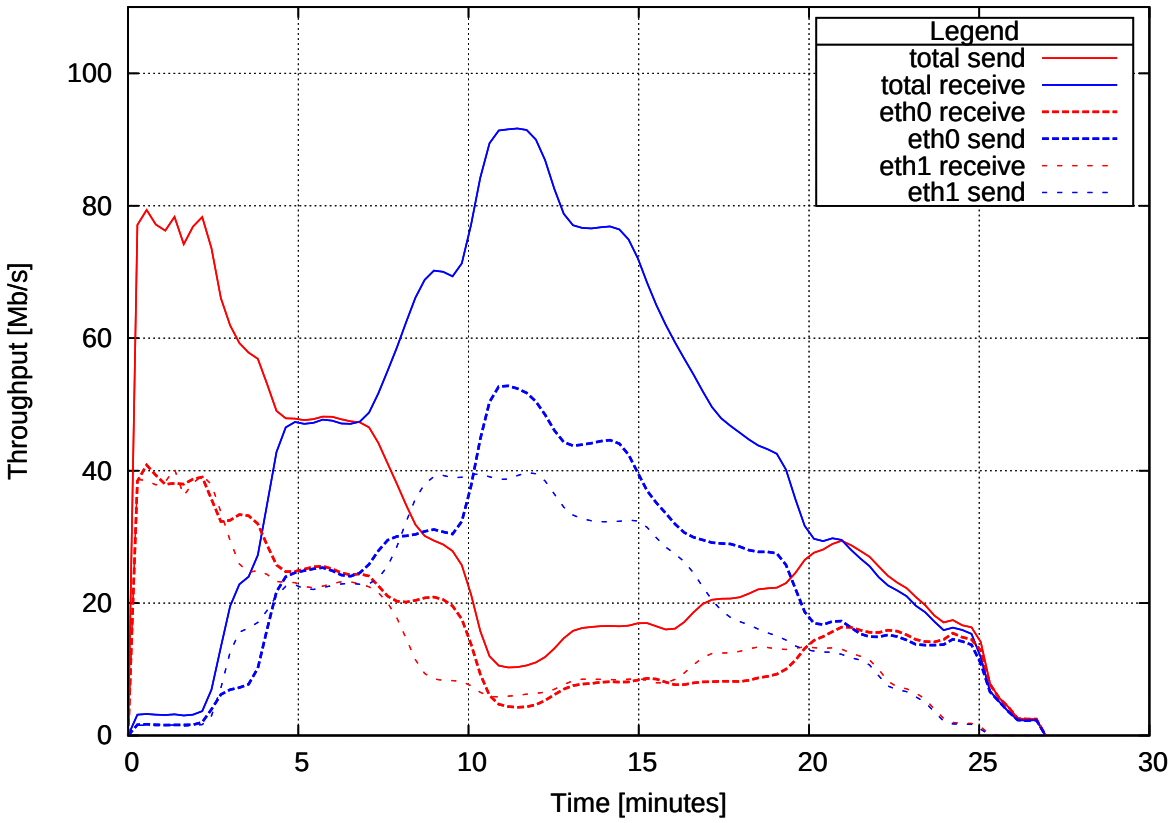


Obrázek 6: Vliv počtů klientů na propustnost.

stupů projektu a provedených testů vyplývá, že potenciální přínos ke zvýšení propustnosti a celkové užitné hodnoty OpenAFS je značný. Na druhou stanu je třeba připomenout, že se jedná o poměrně komplikovanou technologii, která klade další nároky na provoz a administrátory.

V průběhu testování fungovalo rozšíření Rx-OSD naprosto spolehlivě a nebylo zaznamenáno žádné nestandardní chování či výpadek. AFS Rx-OSD je v současnosti rutině používáno v několika velkých AFS buňkách (RZG, DESY,...) a i když má status vývojového rozšíření je velmi stabilní. Na evropské OpenAFS konferenci konané na ZČU[5] bylo ohlášeno, že Rx-OSD bude začleněno do OpenAFS nejpozději ve verzi 1.10 plánované na první čtvrtletí 2011.

[5]European AFS & Kerberos Conference 2010: `http://afs2010.civ.zcu.cz/`.



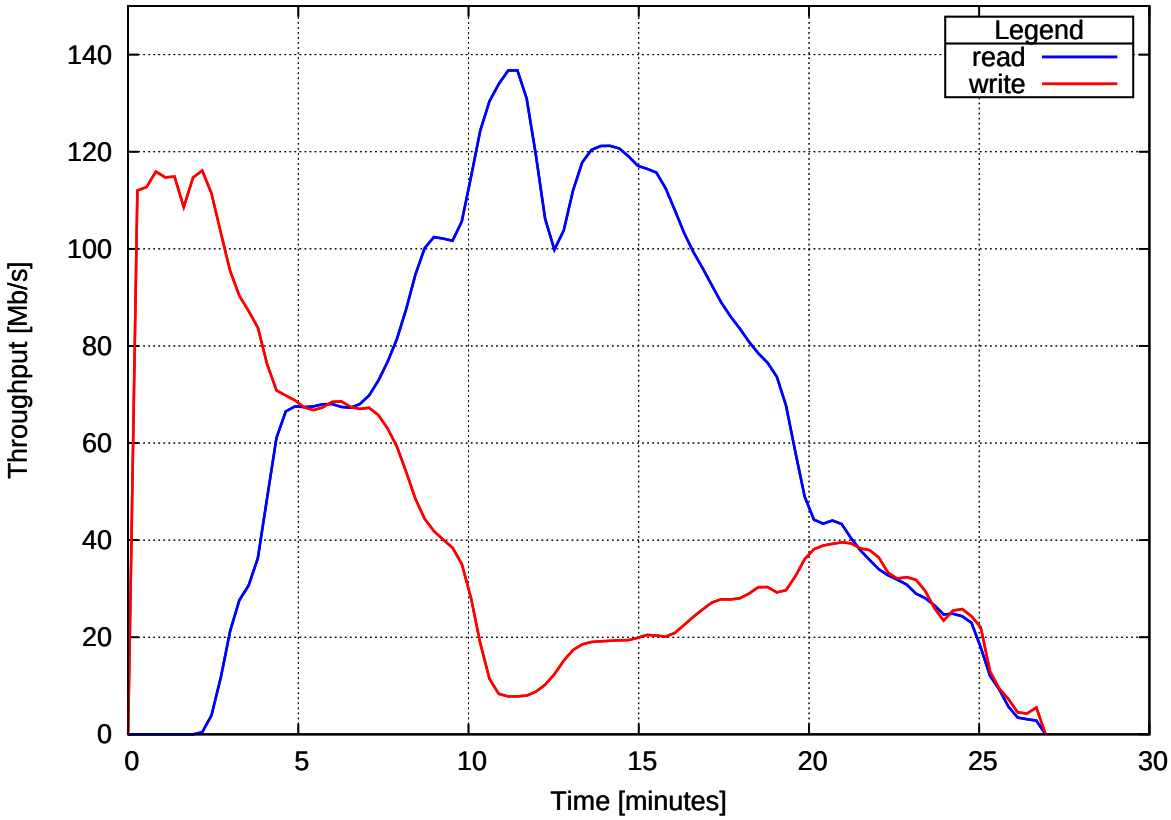
Obrázek 7: Zátěž na síťových rozhraních.

Odkazy

[1] Stránka s elektronickými zdroji projektu
`http://support.zcu.cz/index.php/CIV:Granty/Cesnet_293_2009`

[2] Stránka projektu OpenAFS
`http://www.openafs.org/`

[3] Stránka projektu OpenAFS-OSD
`http://pfanne.rzg.mpg.de/trac/openAFS-OSD`



Obrázek 8: Zátěž disků.