

Xen virtualizācijas sistēmas uzbūve un lietošana

Leo Trukšāns, DatZ030183
Latvijas Universitāte

Kursam "Speciālās nodaļas"

Xen produkts

- Populārākā atklātā pirmkoda virtualizācijas sistēma
 - Iespējams arī komerciāls atbalsts
- Pielietojumi:
 - Serveru konsolidācija
 - Aparātiskā neatkarība
 - Vairāku OS izpilde uz viena datora izstrādei vai testēšanai
 - Kodola izstrādei
 - Klāsteru (*cluster*) skaitļošanai
 - Aparātiskais nodrošinājums modificētām OS

Xen produkts

- Xen sāka veidot Kembridžas universitātē
- Xen 1.0 – 2003. gada oktobrī
- Pašlaik Xen ietilpst Citrix uzņēmumā
- Informācija:
 - <http://www.xensource.com/>
 - <http://en.wikipedia.org/wiki/Xen>
 - <http://www.xen.org/>

Terminoloģija

- VMM – programmatūra, kas ļauj vairākas virtuālās mašīnas multipleksēt vienā fiziskā
- Paravirtualizācija – virtualizācijas metode, kas prasa operētājsistēmas modificēšanu darbam virtuālā mašīnā
- Hipervīzors (*Hypervisor*) – alternatīvs VMM nosaukums – pārvalda "supervīzoros"
- Pilna virtualizācija – virtualizācijas metode, kas neprasa nekādu virtuālās mašīnas sistēmas modificēšanu

Terminoloģija

- *Shadow pagetables* – metode datora atmiņas izvietojuma slēpšanai no virtuālajām mašīnām
- Virtuālā mašīna – vide, kurā strādā mitināta (*hosted*) operētājsistēma un kas piedāvā izdalīta datora abstrakciju
- Domēni – izpildes konteksti, kas satur virtuālo mašīnu
 - Dom0 (Domain 0) – pārvalda sistēmu
 - DomU – viesis
- Domēna ID – unikāls domēna identifikators

Xen būtība

- Paravirtualizējošs virtuālo mašīnu monitors (VMM)
 - Viss darbs ar aparatūru iet caur hipervīzoru
- Īpašs kodols domēnu operētājsistēmām
 - Tam jābūt pielāgotam Xen-am
 - Tam jāspēj strādāt bez procesora 0-tā līmeņa
- Spēj izmantot arī aparātisko virtualizāciju (AMD, Intel), kas stipri vienkāršo darbu ar viesiem un tiek saukta par "asistēto virtualizāciju"
 - Tā var arī palaist MS Windows Linux-ā

Xen iespējas

- Domēni strādā ar minimālu veiktspējas zudumu
 - .. gan skaitļošanas jaudā, gan darbā ar ierīcēm
- Viesu sistēmas "diskam" var izmantot failu vai LVM/EVM sadaļu
 - LVM/EVM sadaļas var kopēt strādājošam domēnam, nodrošinot biežas "karstas" kopijas
- Tipisks viesis satur 4 failus:
 - konfigurācija, kodols, saknes un apmaiņas (*swap*) failsistēma

Xen iespējas

- Viesu sistēmu saglabāšana un migrēšana:
 - Apstādinātā veidā
 - Hibernētā veidā (save/restore)
 - "Dzīvajā" veidā (pārsūta strādājošu sistēmu)
 - 60-300ms laikā, saglabājot atvērtus savienojumus
- Drošības mehānismi domēnu darbības kontrolēšanai
 - sistēmu iebūvētie: ugunsmūri, SELinux, u.c.
 - sHype/Xen, politika

sHype/Xen

- Definē "apzīmējumus" (*labels*), kas nosaka domēna iespējas
 - Pieeju saviem resursiem
 - Izpildīšanas nosacījumi – vai drīkst izpildīt uz šī Xen, un kopā ar kuriem citiem apzīmētajiem domēniem
- Domēnam pieliek apzīmējumu
 - `xm list --label`
 - `xm addlabel ...`
- Drošības politikas definēšanas rīki:
 - `ezPolicy` – grafiskais rīks
 - `xensec_gen` – politikas optimizēšana
 - teksta vai XML redaktors

Viesu plānošana

- Pēc noklusēšanas domēni tiek plānoti (*schedule*) vienmērīgi pa visiem datora procesoriem
- Katru VCPU var "piestiprināt" kādam reālajam CPU
 - xm vcpu-pin
- Viesiem ir svāri (*wight*), kas ļauj proporcionāli sadalīt slodzi
- Viesiem ir arī "cepures" (*cap*), kas norāda, cik procentus no CPU drīkst izmantot
 - xm sched-credit ...

vnet-i

- *vnet* – privāts tīkls domēnu lietošanai
 - līdzīgi VLAN-iem
- Katram vnet-am ir savs konfigurācijas fails
- Manipulēšana ar vnet-iem:
 - xm vnet-create <config>
 - xm vnet-delete <vnetid>
 - xm vnet-list
- vnet-u interfeisi un tilti parādās pie ifconfig un ir letojami domēnu konfigurācijā

Xen instalēšana

- Xen programmatūru un Xen-izētus Linux kodolus var iegūt pakotņu veidā, kā arī kompilēt no pirmkoda
- Instalēšanas gaitā uz bāzes sistēmas tiek uzstādīts:
 - Xen programmatūra (xend, pārvaldības rīki, utml.)
 - Hipervīzors
 - Xen-izēts kodols
 - Linux sāknētājam (grub) uzstādīts Xen-izētas sistēmas palaišanas variants
 - Piezīme: šādu sistēmu var sāknēt ar vai bez Xen

Xen instalēšana

- Lietoju Ubuntu Server 7.04 distributīvu
- Instalēt Xen nodrošinājumu servera kodolam:
 - `sudo aptitude install ubuntu-xen-server`
- `/boot/grub/menu.lst` parādās Xen-izētas sistēmas sāknēšanas ieraksts:
 - `title Xen 3.0-i386-pae / Ubuntu, kernel 2.6.19-4-server`
 - `root (hd0,0)`
 - `kernel /boot/xen-3.0-i386-pae.gz`
 - `module /boot/vmlinuz-2.6.19-4-server ...`
 - `module /boot/initrd.img-2.6.19-4-server`
 - ...

Xen instalēšana

- Xen-izētā sistēmā jāaktivizē xend serviss
 - Ubuntu satur automātiskus skriptus
- Konfigurēt Xen *bridge* iespēju tīklam:
 - `sudo nano /etc/xen/xend-config.sxp`
 - `# (network-script network-dummy)`
 - `(network-script network-bridge)`
- Pēc tā parādās 3 jauni tīkla interfeisi:
 - `ifconfig -s`

```
• Iface  MTU Met  RX-OK RX-ERR RX-DRP RX-OVR  TX-OK TX-ERR TX-DRP TX-OVR Flg
• eth0  1500 0    137   0   0   0    134   0   0   0 BMRU
• lo    16436 0     0   0   0   0     0   0   0   0 LRU
• peth0 1500 0    133   0   0   0    131   0   0   0 BORU
• vif0. 1500 0    134   0   0   0    137   0   0   0 BORU
• xenbr 1500 0     29   0   0   0     0   0   0   0 BORU
```

Viesu izveidošana

- Jāizveido vismaz divi faili:
 - Vieša konfigurācijas fails
 - Direktorijā /etc/xen
 - Failsistēmas attēla fails / sadaļai
 - Ar ext3 vai citu Linux failsistēmu
 - Tā atrašanās vieta nav svarīga
 - Tajā jāiekopē vēlamās operētājsistēmas faili
 - Ieteicams arī viens apmaiņas (*swap*) sadaļas attēla fails

Viesu izveidošana

- `sudo dd if=/dev/zero of=/usr/local/serv1.ext3 bs=1024k seek=700 count=0`
- `sudo mkfs.ext3 /usr/local/serv1.ext3`
- `sudo mkdir /mnt/serv1`
- `sudo mount /usr/local/serv1.ext3 /mnt/serv1 -o loop`
- `sudo debootstrap feisty /mnt/serv1 http://lv.archive.ubuntu.com/ubuntu`
- `sudo cp -a /lib/modules/2.6.19-4-server/ /mnt/serv1/lib/modules/`

Viesu izveidošana

- `sudo nano /mnt/serv1/etc/network/interfaces`
 - `auto lo`
 - `iface lo inet loopback`
 -
 - `auto eth0`
 - `iface eth0 inet dhcp`
- `sudo cp /etc/hosts /mnt/serv1/etc/hosts`
- `sudo nano /mnt/serv1/etc/hostname`
- `sudo nano /mnt/serv1/etc/fstab`
 - `proc /proc proc defaults 0 0`
 - `/dev/sda1 / ext3 defaults,errors=remount-ro 0 1`
- `sudo umount /mnt/serv1`

Viesu izveidošana

- `sudo nano /etc/xen/serv1.cfg`
 - `kernel = "/boot/vmlinuz-2.6.19-4-server"`
 - `ramdisk = "/boot/initrd.img-2.6.19-4-server"`
 - `builder='linux'`
 - `memory = 128`
 - `name = "serv1"`
 - `vcpus = 1`
 - `vif = ['bridge=xenbr0']`
 - `disk = ['file:/usr/local/serv1.ext3,ioemu:sda1,w']`
 - `root = "/dev/sda1 ro"`
- Noteiktai MAC adresei:
 - `vif = ['bridge=xenbr0, mac=00:16:3e:01:02:03']`
- Lai domēns tiktu aktivizēts datora sāknēšanā, tā konfigurācijas failu (vai saiti) ieliek direktorijā `/etc/xen/auto/`

Viesu izveidošana

- Domēnam tiks izdalīta prasītā operatīvā atmiņa
- Konfigurācijā var norādīt tieši izmantojamās PCI ierīces parametrus
 - Tas ļauj atdot domēnam kādu ierīci
 - Lielāks ātrums darbam ar ierīci
 - Var izmantot īpašus dziņus viesī
 - Bet ekskluzīvas tiesības uz ierīci
- Failsistēmai var izmantot arī NFS eksportu

Viesu darbināšana

- Apskatīt atvērto domēnu sarakstu:
 - `sudo xm list`
- Aktivizēt viesi:
 - `sudo xm create /etc/xen/serv1.cfg`
- Aktivizēt viesi ar konsoles piesaisti:
 - `sudo xm create /etc/xen/serv1.cfg -c`
- Pievienoties vieša konsolei:
 - `sudo xm console serv1`
- Atvienoties no konsoles – `Ctrl+5`
- Beidzēt viesi (*shutdown*):
 - `sudo xm shutdown serv1`
- Pārtraukt viesi ("atvienot elektrību"):
 - `sudo xm destroy serv1`

Viesu sākotnējais stāvoklis

- Sākumā ir tukša root parole. Tādēļ piesakamies ar root un uzstādam paroli:
 - passwd root
- Apskatamies aizņemto diska vietu un operatīvo atmiņu:
 - df – ~300MB izmantoti uz / (Ubuntu Feisty)
 - top – ~28MB izmantoti (Ubuntu Feisty)
- Tīklam jau jāstrādā ar uzstādīto konfigurāciju:
 - ifconfig – eth0 ar definēto MAC adresi
 - ping www.lumii.lv

Viesu konfigurēšana (Ubuntu)

- Aizpildam APT avotu failu:
 - vi /etc/apt/sources.list
 - deb http://lv.archive.ubuntu.com/ubuntu feisty main restricted
 - deb-src http://lv.archive.ubuntu.com/ubuntu/ feisty main restricted
 -
 - deb http://lv.archive.ubuntu.com/ubuntu/ feisty-updates main restricted
 - deb-src http://lv.archive.ubuntu.com/ubuntu/ feisty-updates main restricted
 -
 - deb http://lv.archive.ubuntu.com/ubuntu/ feisty universe
 - deb-src http://lv.archive.ubuntu.com/ubuntu/ feisty universe
 -
 - deb http://lv.archive.ubuntu.com/ubuntu/ feisty multiverse
 - deb-src http://lv.archive.ubuntu.com/ubuntu/ feisty multiverse
 -
 - deb http://security.ubuntu.com/ubuntu feisty-security main restricted
 - deb-src http://security.ubuntu.com/ubuntu feisty-security main restricted
 - deb http://security.ubuntu.com/ubuntu feisty-security universe
 - deb-src http://security.ubuntu.com/ubuntu feisty-security universe
 - deb http://security.ubuntu.com/ubuntu feisty-security multiverse
 - deb-src http://security.ubuntu.com/ubuntu feisty-security multiverse
- Atjauninam programmatūru:
 - aptitude update
 - aptitude upgrade

Viesu papildināšana

- Šinī posmā viesu sistēmās ieinstalējam un noskaņojam vajadzīgo programmatūru
- Par piemēru uzstādīsim kādu servera pakotni
 - aptitude install lighttpd
- Tā aizņem maz vietas un atmiņas
- Tās faili ir zem /var/www/, bet žurnāls
 - /var/log/lighttpd/access.log

Viesu "iesaldēšana"

- Efekts ir līdzīgs, kā hibernējot datoru – operatīvās atmiņas saturs tiek saglabāts, šoreiz – failā
- Iesaldējam:
 - `sudo xm save serv1 save1`
- Atjaunojam:
 - `sudo xm restore save1`
- Faili: `save1`, `serv1.ext3`, `serv1.cfg`
- Eksperiments – ar viesī strādājošu ping

Viesu "iesaldēšana" un pārceļšana uz citu datoru

- Uz pirmā datora (10.1.2.11):
 - sudo xm save serv1 save1
 - scp save1 user@10.1.2.19:~/
 - scp /usr/local/serv1.ext3 user@10.1.2.19:~/
 - scp /etc/xen/serv1.cfg user@10.1.2.19:~/
- Uz otrā datora (10.1.2.19):
 - sudo chown root:root serv1.ext3
 - sudo chown root:root serv1.cfg
 - sudo mv serv1.ext3 /usr/local/
 - sudo mv serv1.cfg /etc/xen
 - sudo xm restore save1
- Piezīme: serv1.cfg jākopē tikai vienreiz

Viesu migrēšana tiešsaistē

- Kā save/restore, tikai informāciju nodod tieši otram datoram
 - Pie tam sūta tikai izmaiņas, tā ātrāk
 - Sūta īpašam Xen procesam – *relocation service*
- Komutatora MAC tabulas jautājums
 - Risinājuma idejas: ar ping-iem, īpašu ARP
- Aizturu jautājums
 - Vai domēns būs nepieejams mazāk par 300ms?
- Eksperiments – ar ping un atvērtu ssh sesiju
 - Šinī eksperimentā – divi pingi neatsaucās (~2s)
 - Apmēram 10-15s pirms pārrāvuma bija liela tīkla aktivitāte (sītīja domēnu otram datoram)

Paldies par uzmanību!

Jautājumi?

Tēze

- Xen viesu sistēmās var lietot ārkārtīgi kompaktas operētājsistēmas
 - Arī tādas, kurās atstāts tikai veicamajam uzdevumam nepieciešamais funkciju klāsts
- Īpaši optimizētu Xen videi operētājsistēmas attelu kopā ar kodolu saukšu par *minimašīnu*
- Tās katra pilda noteiktu *Uzdevumu*
- Linux sniedz lielas iespējas tādu veidošanai

Minimašīnu priekšrocības

- Kompaktums
- Izplatāmība
- Aizvietojamība

Minimašīnas uzbūve

- Optimizēts kodols
 - Sākam ar standarta Ubuntu Xen kodolu – 1,5MB fails, ~20MB atmiņas
- Minimizēts dziņu komplekts
 - Atstāj tikai virtuālo ierīču dziņus, visi īstie aparatūras dziņi paliek tikai saimnieka sistēmai
- Tikai tās programmu un bibliotēku pakotnes, kas nepieciešamas Uzdevumam
- Optimizēta sāknēšana
 - Bez konsolēm utml.

Paldies par uzmanību!

Jautājumi?