



ОС OpenSolaris: ТЕХНИЧЕСКИЕ ВОЗМОЖНОСТИ

Сергей Пикалёв
Sun Microsystems

Немного истории

- 4 марта 2005 года вышел Solaris 10
- 14 июня 2005 года запущен проект OpenSolaris
- Лето 2007 года – старт проекта Indiana

Solaris и OpenSolaris

- Solaris 10
- Nevada
- Indiana

Что нового в OpenSolaris?

- Управление сервисами (smf)
- Управление ресурсами (rmf)
- Трассировка и профилировка (Dtrace)
- Виртуализация
 - > Контейнеры
 - > Гипервизор xVM
 - > Виртуальная машина VirtualBox
- Файловая система zfs

Простота администрирования

- Унифицированная система команд
 - > *cfg – настройка
 - > *adm – управление
 - > *ctl - “горячая” настройка
- Автодополнение с подсказками

smf: управление сервисами

- Единый механизм настройки и управления
- Замена старых механизмов
 - > inetd.conf
 - > stand-alone сервисы
- Новые свойства
 - > аналог registry
 - > восстановление после сбоев

smf: управление сервисами

Практика

- Команды
 - > Список сервисов: **svcs**
 - > Настройка: **svccfg**
 - > Администрирование: **svcadm**

Примеры:

```
$ svcs -a | grep physycal
```

```
$ svcadm disable nwam
```

rmf: управление ресурсами

- Объекты управления
 - > память, диск, процессы, процессор и т.д.
- Типы распределения
 - > ограничение
 - > разделение
 - > выделение гарантированной доли
- Проекты и задания

rmf: проекты и задания

Практика

- Команды
 - > Получение информации: `projects`
 - > Управление: `projadd`, `projmod`, `projdel`
 - > Запуск: `newtask`
- Файлы
 - > `/etc/project`

Примеры:

```
$ projadd -U sep testprj
```

```
$ newtask -p testp
```

rmf: атрибуты и настройка

- Структура
 - > Триплеты (уровень, значение, действие)
 - > Уровни: **system, privileged, basic**
 - > Действия: **none, deny, signal**

rmf: атрибуты и настройка

Практика

- Команды
 - > `prctl`

Примеры:

```
$ prctl -i project testprj
```

```
$ projmod -a -K "process.max-file-size=(basic,1000000,deny)" testprj
```

```
$ prctl -r -t basic -n process.max-file-size -v 1024 -e signal=KILL -i process <pid>
```

Dtrace: динамическая трассировка

- Побудительные мотивы
 - > наличие труднодиагностируемых проблем
 - > дороговизна статического инструментирования программ и ядра
 - > увеличение времени прогона программы влияет на воспроизводимость проблем

Dtrace: динамическая трассировка

- Динамическое инструментирование
 - > не требуется перекомпиляция
- Динамическое управление точками трассировки (on/off)
- Единый инструмент
 - > профилирование, отладка, изучение
 - > приложения, библиотеки, ядро
- Безопасность
 - > нет угрозы падения приложения или ядра

Dtrace: динамическая трассировка

Практика

- Структура датчика
 - > провайдер:модуль:функция:имя
- Команды
 - > **dtrace**

Примеры:

```
syscall::write:entry
```

```
{  
    @cnt[execname, uid]=count()  
}
```

Контейнеры

Система виртуализации

- Контейнер = зона + управление ресурсами
- Основная концепция: изолированная среда исполнения в рамках одного экземпляра ОС
- Компоненты:
 - > Зоны (изоляция пространства имен)
 - > Управление ресурсами (процессор, память, дисковое пространство и т.д.)

Зоны: объекты виртуализации

- Файловая система
- Устройства
- Сеть
- Процессы

Зоны: файловая система

- Своя корневая система (/)
- Файловые системы могут быть
 - > унаследованы (“ro”)
 - > скопированы
 - > смонтированы (как “ro” так и “rw”)
- Умолчания
 - > /usr, /lib, /sbin, /platform – унаследованы
 - > /etc, /opt – скопированы
- Особенности реализации
 - > NFS-сервер в зоне не работает

Зоны: устройства

- Зоны видят набор безопасных псевдоустройств в /dev
 - > /dev существует, /devices – нет
 - > устройства /dev/random и /dev/console безопасны, /dev/ip – нет
- Добавление устройства в зону
 - > zonectl: myzone> add device
- Зоны могут модифицировать атрибуты существующих устройств, но не могут делать mknod (2)

Зоны: сеть

- Зона может иметь собственные
 - > Ipv4/IPv6 адреса
 - > имя хоста
 - > пространство портов
 - > сервис имен
- Зона не может видеть трафик других зон

Зоны: процессы

- Каждый процесс связан с одной зоной
- Некоторые системные вызовы запрещены или ограничены внутри зон
- Процессы внутри зоны взаимодействуют как обычно
- Процессы внутри зоны не видят процессов из других зон
 - > /proc виртуализирована
- Из глобальной зоны доступны все процессы

Зоны

Практика

- Команды
 - > Создание и настройка: **zonectfg**
 - > Управление: **zoneadm**
 - > Вход: **zlogin**

Примеры:

```
$ zoneadm list -iv
```

```
$ zonectfg -z newzone
```

```
$ zoneadm -z newzone install
```

```
$ zlogin -C
```

```
$ zoneadm -z newzone boot
```

ZFS

- “Трудные места” файловых систем
 - > “тихие” повреждения данных
 - > накладные расходы на журналирование
 - > проблемы переносимости между платформами с разным порядком байтов
 - > физические ограничения
 - размер диска/тома/файла
 - количество файлов/блоков/узлов

ZFS: основные свойства

- Скорее память нежели диск
- Транзакционная модель “copy-on-write”
 - > “бесплатный” журнал
 - > “бесплатные” срезы и резервные копии
- Новые объемы добавляются в пул
- Все операции “прикрыты” контрольными суммами
- Первая 128-битная файловая система
- Не зависит от порядка байтов

ZFS

Практика

- Команды
 - > Конфигурация: `zpool`
 - > Управление: `zfs`

Примеры:

```
$ zpool create newpool c1t0
```

```
$ zpool add newpool c2t0
```

```
$ zfs list
```

```
$ zfs create newpool/bin
```

Что еще?

- Бренд-зоны
- Система прав доступа (RBAC)
- Отладка ядра (mdb)
- Управление сетевыми ресурсами
 - > Проект CrossBow
- Расширенный доступ к файлам (ACL)



ОС OpenSolaris: ТЕХНИЧЕСКИЕ ВОЗМОЖНОСТИ

Сергей Пикалёв
Sergei.Pikalev@Sun.com