What's new in Solaris 10?

Derek Morr
Senior Systems Programmer
Emerging Technologies
Release History

Jan 02
Build 1

Apr 02
Internal Product Use

Apr 03
Process Rights Mgmt

Sept 03
Software Express

Oct 03
FireEngine

Nov 03
DTrace ; BART

Feb 04
Solaris ipf

Mar 04
Entered Beta

July 04
SCTP; DTrace IO

Sept 03
NFSv4; Cryptographic Framework

Aug 03
NFSv4; Cryptographic Framework

Oct 03
NFSv4; Cryptographic Framework

Nov 03
DTrace ; BART

Feb 04
Solaris ipf

Mar 04
Entered Beta

June 04
Predictive Self-healing

Aug 04
SATA; x86 VTS; KMDB

Nov 04
AMD64

Nov 04
Solaris 10 Released

Jan 05
Solaris 10 Released
improved x86 support (2x HCL; X.org)
logging enabled by default on all filesystems
devfs for /devices
microstate accounting always on
New Features

- Zones
- DTrace
- Fire Engine
- Services
- ipf-based firewall
- AMD64
- Java Desktop System
Software-partitioning of a system

Isolated system image (up to 8,192 zones)

Migration

Virtualizes hardware resources

Single kernel (the global zone); partially shared storage

Server consolidation, development, security
Child Zone limits

- can’t ifconfig or alter routing table
- no raw sockets
- can’t make child zones
- no DTrace
- no NFS server
- no /devices
Creating a zone

mkdir /zones/myZone
chmod 700 /zones/myZone
zonecfg -z myZone
zoneadm -z myZone install
zoneadm -z myZone boot
<table>
<thead>
<tr>
<th>PID</th>
<th>USERNAME</th>
<th>USR</th>
<th>SYS</th>
<th>TRP</th>
<th>TFL</th>
<th>DFL</th>
<th>LCK</th>
<th>SLP</th>
<th>LAT</th>
<th>VCX</th>
<th>ICX</th>
<th>SCL</th>
<th>SIG</th>
<th>PROCESS/NLWP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1619</td>
<td>derek</td>
<td>0.2</td>
<td>0.8</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>99</td>
<td>0.0</td>
<td>30</td>
<td>0</td>
<td>190</td>
<td>0</td>
<td>0</td>
<td>prstat/1</td>
</tr>
<tr>
<td>1590</td>
<td>root</td>
<td>0.1</td>
<td>0.1</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>29</td>
<td>0</td>
<td>113</td>
<td>0</td>
<td>0</td>
<td>nmap/1</td>
</tr>
<tr>
<td>1265</td>
<td>derek</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>2</td>
<td>0</td>
<td>24</td>
<td>0</td>
<td>0</td>
<td>sshd/1</td>
</tr>
<tr>
<td>199</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>1</td>
<td>0</td>
<td>5</td>
<td>1</td>
<td>0</td>
<td>xntpd/1</td>
</tr>
<tr>
<td>623</td>
<td>derek</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>77</td>
<td>23</td>
<td>0</td>
<td>34</td>
<td>0</td>
<td>19</td>
<td>0</td>
<td>java/13</td>
</tr>
<tr>
<td>619</td>
<td>derek</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>73</td>
<td>27</td>
<td>0</td>
<td>35</td>
<td>0</td>
<td>19</td>
<td>0</td>
<td>java/15</td>
</tr>
<tr>
<td>78</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>35</td>
<td>0</td>
<td>97</td>
<td>0</td>
<td>0</td>
<td>nscl/25</td>
</tr>
<tr>
<td>528</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>ttymon/1</td>
</tr>
<tr>
<td>572</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>htt_server/2</td>
</tr>
<tr>
<td>423</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>zsched/1</td>
</tr>
<tr>
<td>334</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>htt_server/2</td>
</tr>
<tr>
<td>297</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>secd/2</td>
</tr>
<tr>
<td>398</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>vold/2</td>
</tr>
<tr>
<td>421</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>devfsadm/6</td>
</tr>
<tr>
<td>172</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>utmpd/1</td>
</tr>
<tr>
<td>418</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>zoneadmd/4</td>
</tr>
<tr>
<td>219</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>73</td>
<td>27</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>syslogd/15</td>
</tr>
<tr>
<td>168</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>ttymon/1</td>
</tr>
<tr>
<td>211</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>ttymon/1</td>
</tr>
<tr>
<td>82</td>
<td>daemon</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>kcfd/3</td>
</tr>
<tr>
<td>384</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>40</td>
<td>60</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>auditd/5</td>
</tr>
<tr>
<td>90</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>powerd/3</td>
</tr>
<tr>
<td>120</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>sshd/1</td>
</tr>
<tr>
<td>59</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>71</td>
<td>29</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>syseventd/14</td>
</tr>
<tr>
<td>206</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>sac/1</td>
</tr>
<tr>
<td>66</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>17</td>
<td>83</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>picld/6</td>
</tr>
<tr>
<td>314</td>
<td>root</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>100</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>htt/1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ZONEID</th>
<th>NPROC</th>
<th>SIZE</th>
<th>RSS</th>
<th>MEMORY</th>
<th>TIME</th>
<th>CPU</th>
<th>ZONE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>42</td>
<td>194M</td>
<td>106M</td>
<td>5.3%</td>
<td>0:00:53</td>
<td>0.5%</td>
<td>global</td>
</tr>
<tr>
<td>1</td>
<td>20</td>
<td>301M</td>
<td>139M</td>
<td>7.0%</td>
<td>0:02:52</td>
<td>0.0%</td>
<td>saslca</td>
</tr>
</tbody>
</table>

Total: 62 processes, 245 lwps, load averages: 0.01, 0.00, 0.00
Solaris supports per-zone resource limits:
- CPU time
- RAM usage
- Disk usage
- Number of LWPs
Zones Demo
Traditional Solaris observability tools: kstat, prstat, lockstat, busstat, cpustat, trapstat, iostat, mpstat, netstat, nfsstat, vmstat

Produce a lot of data

Have to manually correlate

Difficult to observe system-wide
DTrace is a new dynamic tracing framework in Solaris 10.

- Doesn’t use static tracing points.
- DTrace providers dynamically insert trace points when activated.
- Zero probe effect when not activated.
- Data can be filtered and aggregated by the probes – greatly reducing output.
New language, D (based on C), allows you to monitor any probe in the system

As of 11/04:
- 41,181 probes on SPARC
- 44,915 on x86

Available on x86(-64) and SPARC

Open sourced (CDDL) in January 05
Probe Families

- lockstat – kernel locks
- proc
- vminfo
- sysinfo
- io
- mib
- profile
- fbt – function boundary tracing
- sched
- pid
- fpuinfo
D is based on ANSI C, with data structures

Runs in the D virtual machine in the kernel

Probes only enabled when needed

Uses Least Privilege for safety

Users restricted to dtracing their own procs
Who’s causing pageins?

```sh
#!/usr/sbin/dtrace -s
vminfo:::pgin
{
  @swappers[pid, execname] = count();
}
```
D Examples

<table>
<thead>
<tr>
<th>Process ID</th>
<th>Command</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>6829</td>
<td>bash</td>
<td>1</td>
</tr>
<tr>
<td>192</td>
<td>nscd</td>
<td>3</td>
</tr>
<tr>
<td>6837</td>
<td>gfind</td>
<td>3</td>
</tr>
<tr>
<td>6840</td>
<td>bash</td>
<td>9</td>
</tr>
<tr>
<td>6838</td>
<td>sort</td>
<td>10</td>
</tr>
<tr>
<td>6837</td>
<td>gupdatedb</td>
<td>24</td>
</tr>
<tr>
<td>6840</td>
<td>sudo</td>
<td>30</td>
</tr>
<tr>
<td>6854</td>
<td>gzip</td>
<td>13434</td>
</tr>
</tbody>
</table>
Who's doing IO?

```bash
#!/usr/sbin/dtrace -s
io:::start
{
  @[execname, args[2]->fi_pathname] = sum(args[0]->b_bcount);
}
```
D Examples

<table>
<thead>
<tr>
<th>Command</th>
<th>Path</th>
<th>Size (Bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>sched</td>
<td>/home/eschrock/.dt/sessionlogs/machine_DISPLAY=:0</td>
<td>4096</td>
</tr>
<tr>
<td>xlp</td>
<td>/var/adm/utmpx</td>
<td>4096</td>
</tr>
<tr>
<td>fsflush</td>
<td>/export/iso/solaris_4.iso</td>
<td>73728</td>
</tr>
<tr>
<td>sched</td>
<td>&lt;none&gt;</td>
<td>82432</td>
</tr>
<tr>
<td>cp</td>
<td>&lt;none&gt;</td>
<td>114688</td>
</tr>
<tr>
<td>fsflush</td>
<td>&lt;none&gt;</td>
<td>177152</td>
</tr>
<tr>
<td>cp</td>
<td>/export/iso/solaris_4.iso</td>
<td>238936064</td>
</tr>
<tr>
<td>cp</td>
<td>/export/iso/solaris_1.iso</td>
<td>239910912</td>
</tr>
</tbody>
</table>
Who’s generating network traffic?

#!/usr/sbin/dtrace -s
mib:::tcpOutDataBytes
{
    @[execname] = sum(args[0]);
}

D Examples
A more involved example

syscall::write:entry
/execename == "sshd"
{
    @[arg0] = quantize(arg2);
}
The D Language

- Strings are a real datatype
- Associative arrays
- Garbage collection
- Thread-local storage
Useful functions

- tracemem() – captures memory
- stack() – records kernel stack trace
- ustack() – records user stack trace
- quantize() – quantize the captured data
DTrace Demo
Part of Predictive Self-Healing

Services are named

```
svc:/network/sshd:default
```

Services managed by `svc.startd`

Configuration managed by `svc.configd`

XML files in `/var/svc/manifest` describe services

Legacy `/etc/rc*.d` scripts still work
SMF benefits

- New observability tools
- Automatic restarting of failed services
- Parallel startup
- Improved diagnostic tools (`svcs -x`)
- Quieter boot
- All logs in `/var/svc/logs` or `/etc/svc/volatile`
<service name='system/coreadm' type='service' version='1'>
    <single_instance />
    <instance name='default' enabled='true'>
        <dependency name='configuration' grouping='require_all' restart_on='none' type='path'>
            <service_fmri value='file:///etc/coreadm.conf' />
        </dependency>
        <exec_method type='method' name='start'
            exec='/lib/svc/share/bin/svc-coreadm %f' timeout='0' />
        <exec_method type='method' name='stop'
            exec=':true' timeout='0' />
    </instance>
    <stability value='Evolving' />
    <template>
        <description><local_description locale='C'>
            System-wide core file configuration service.
        </local_description></description>
        <documentation>
            <manpage title='coreadm' section='1M' manpath='/usr/share/man' />
        </documentation>
    </template>
</service>
What services depend on the network?

derek@interzone derek $ svcs -D network/physical

<table>
<thead>
<tr>
<th>STATE</th>
<th>STIME</th>
<th>FMRI</th>
</tr>
</thead>
<tbody>
<tr>
<td>disabled</td>
<td>Feb_01</td>
<td>svc:/network/dns/client:default</td>
</tr>
<tr>
<td>disabled</td>
<td>Feb_01</td>
<td>svc:/network/inetd:default</td>
</tr>
<tr>
<td>disabled</td>
<td>Feb_01</td>
<td>svc:/network/nfs/client:default</td>
</tr>
<tr>
<td>disabled</td>
<td>Feb_01</td>
<td>svc:/network/rarp:default</td>
</tr>
<tr>
<td>disabled</td>
<td>Feb_01</td>
<td>svc:/network/slp:default</td>
</tr>
<tr>
<td>online</td>
<td>Feb_01</td>
<td>svc:/system/identity:domain</td>
</tr>
<tr>
<td>online</td>
<td>Feb_01</td>
<td>svc:/system/identity:node</td>
</tr>
<tr>
<td>online</td>
<td>Feb_01</td>
<td>svc:/milestone/single-user:default</td>
</tr>
<tr>
<td>online</td>
<td>Feb_01</td>
<td>svc:/network/initial:default</td>
</tr>
<tr>
<td>online</td>
<td>Feb_07</td>
<td>svc:/network/nfs/status:default</td>
</tr>
<tr>
<td>online</td>
<td>Feb_07</td>
<td>svc:/network/nfs/nlockmgr:default</td>
</tr>
<tr>
<td>online</td>
<td>Feb_07</td>
<td>svc:/network/nfs/server:default</td>
</tr>
</tbody>
</table>
What services does the NFS server depend on?

derek@interzone derek $ svcs -d nfs/server

<table>
<thead>
<tr>
<th>STATE</th>
<th>STIME</th>
<th>FMRI</th>
</tr>
</thead>
<tbody>
<tr>
<td>disabled</td>
<td>Feb_01</td>
<td>svc:/network/rpc/keyserv:default</td>
</tr>
<tr>
<td>online</td>
<td>Feb_01</td>
<td>svc:/network/loopback:default</td>
</tr>
<tr>
<td>online</td>
<td>Feb_01</td>
<td>svc:/system/filesystem/local:default</td>
</tr>
<tr>
<td>online</td>
<td>Feb_06</td>
<td>svc:/network/physical:default</td>
</tr>
<tr>
<td>online</td>
<td>Feb_07</td>
<td>svc:/network/rpc/bind:default</td>
</tr>
<tr>
<td>online</td>
<td>Feb_07</td>
<td>svc:/network/nfs/nlockmgr:default</td>
</tr>
<tr>
<td>online</td>
<td>Feb_07</td>
<td>svc:/network/nfs/mapid:default</td>
</tr>
<tr>
<td>uninitialized</td>
<td>Feb_01</td>
<td>svc:/network/rpc/gss:default</td>
</tr>
<tr>
<td>uninitialized</td>
<td>Feb_01</td>
<td>svc:/network/rpc/gss:ticotsord</td>
</tr>
</tbody>
</table>
Run levels implemented as milestones:

- S - single-user
- 2 - multi-user
- 3 - multi-user-server
Get Solaris 10
http://www.sun.com/software/solaris/get.jsp

Sun Bloggers
http://blogs.sun.com/

OpenSolaris
http://www.opensolaris.org/
Resources

Zones

http://www.sun.com/bigadmin/content/zones/

http://www.sun.com/bigadmin/features/articles/solaris_zones.html

http://www.blastwave.org/docs/Solaris-10-b51/DMC-0002/dmc-0002.html
DTrace

http://www.sun.com/bigadmin/content/dtrace/

http://blogs.sun.com/roller/page/bmc/20040805#demo_ing_dtrace
