Tracing I/O USING   
**BLOCK TRACE**

**Getting Started with I/O Trace Implementation**

**Using BlockTrace**

**Objectives**

In this tutorial, you will learn how to install a blocktrace and how to captures all activity in the storage stack in the Linux kernel’s block layer from the OS perspective.

|  |  |
| --- | --- |
| **Tutorial Action** | **Time Required(approx)** |
| **Step 1: Install Blocktrace in your Linux machine** You will learn how to setup a **BlockTrace** tool and captures all activity in the storage stack in the Linux kernel’s block layer from the OS perspective | 3 minutes |
| **Step 2: Install “seekwatcher” and get output image/movie.**  You will learn how to setup seekwatcher to visualize block I/O patterns. | 1 minute |
| **Step 3: Run Blocktrace and Create a file using “dd” that will do disk access** In this step you will learn how to make a disk access | 1 minute |
| **Step 4: Run Blockparse and Seekwatcher to visualize the block I/O pattern** Lastly, trace the I/O patterns | 5 minutes |

**Prerequisites**

The following prerequisites are expected for successful completion of this tutorial:

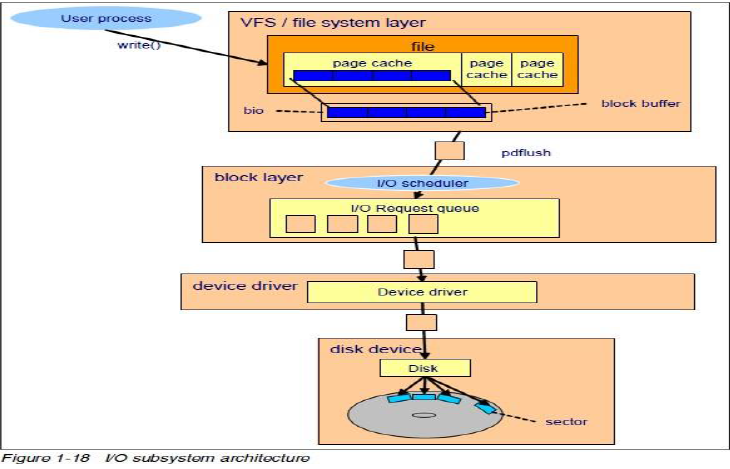
* An SSH client (Windows users: download PuTTY from here: <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>)
* A basic knowledge of Linux
* Install Blocktrace and Seekwatcher.

**Blocktrace :**

* Provides an ability to see the events going inside block i/o layer.
* Using it, it is possible to generate events for all I/O request and monitor it from beginning.
* It has low overhead, less that 2%.

**Architecture level :**

The following diagram illustrates the the level at which blktrace captures the IO workload. Blktrace captures events up to the driver level and operates one level higher than a bus trace does.



**Why blocktrace?**

It can provide a valuable way to test multi-drive workload, because the trace occurs at the logical layer prior to the physical layer. By tracing the logical layer all drives can be traced at once, where a bus trace would be limited by drive numbers. Every operating system is logging all the access its doing and so one can get a pretty broad picture of accesses.



**Events Captured :**

|  |  |  |
| --- | --- | --- |
| **Action** |  | **Description** |
| A | **Remap** | : I/O was remapped to different device. |
| B | **Bounced** | : The data pages attached to this bio are not reachable by the hardware and  must be bounced to a lower memory location. |
| C | **Completion** | : I/O Completion. |
| D | **Issued** | : I/O issued to driver. |
| F | **Front merge** | : I/O ends where a previously inserted requests starts |
| G | **Get request** | : To send any type of request to a block device, a struct request container must  be allocated first |
| I | **Inserted** | : A request is being sent to the i/o scheduler for addition to the internal queue  and later service by the driver. |
| M | **Back merge** | : A previously inserted request exists that ends on the boundary of where this  I/O begins, so the i/o scheduler can merge them together. |
| P | **Plug** | : When i/o is queued to a previously empty block device queue |
| Q |  | : I/O request handled by queue code. |
| S | **Sleep** | : No available request structures were available, so the issuer has to wait for  one to be freed. |
| T | **Unplug(timer)** | : unplug due to timer |
| U | **Unplug** | : Some request data already queued in the device, start sending requests to the driver. |
| X | **Split** | : An incoming I/O may straddle a device or internal zone and needs to be  chopped up into smaller pieces for service. |

**Hands-on**

**Step 1: Install Blocktrace in your Linux machine**

In this step, you will –Install Block trace, if not there.

root@rack1-1:~# apt-get install blktrace

**Step 2: Install “seekwatcher” and get output image/movie.**

In this step, you will –Install seekwatcher, if not there.

root@rack1-1:~# apt-get install seekwatcher

**Step 3: Run Blocktrace and Create a file using “dd” that will do disk access**

root@rack1-1:~# blktrace /dev/sda &

root@rack1-1:~# dd if=/dev/zero of=abc.bin bs=1MB count=1000

NOTE : kill the blktrace process using process id and then run blockparse

root@rack1-1:~# kill (pid)

**Step 4: Run block parse and seekwatcher to visualize block i/o pattern.**

In this step, you will –Use blkparse and seekwatcher to visualize I/O patterns.

Blkparse : In this step you can monitor the drive you want and also pipe the selected output in text file.

root@rack1-1:~# blkparse –i sda.blktrace.7 –f “%a,%S,%t\n” | sed –n ‘/[CD]/ p’ > result.txt

Seekwatcher : In this step, it will give you .png format image of disk IO, Throughput, Seek Count, IOPs

root@rack1-1:~# seekwatcher --io-graph-marker-size=5 -t tracenew5.blktrace.8 -o new5-dd1.png --dpi 200