Blivet analysis

Tao Wu

January 27, 2015

1 Introduction

Blivet provides an easy access to disk management, including partitioning on disks, making format such as ext4,ext3 or xfs on partitions, creating pv/vg/lv and so on. This article focus on the internal implement of blivet, which will help users to locate source of bugs more quickly.

A normal work flow of blivet is as follows:

- Collect device information from udev and build a device tree
- Get info from device tree or change device tree and create actions
- Use partition algorithm to do partition inside device tree
- Process actions in device tree so that changes can be written into disks

2 Build device tree

The device tree is created through populate() method: 1. set maximal waiting time to 300 secends for udev to be done. 2. addUdevDevice add udevXXXdevice 3. handleUdevDeviceFormat handleUdevXXXFormat

2.1 libudev intro

2.2 devices intro

2.3 formats intro

2.4 devicetree structure

get devices:
def getDeviceBySysfsPath(self, path, incomplete=False, hidden=False):
def getDeviceByUuid(self, uuid, incomplete=False, hidden=False):
def getDevicesBySerial(self, serial, incomplete=False, hidden=False):
def getDeviceByLabel(self, label, incomplete=False, hidden=False):
def getDeviceByName(self, name, incomplete=False, hidden=False):

def getDeviceByPath(self, path, preferLeaves=True, incomplete=False, hidden=False):
def getDevicesByType(self, device_type):
def getDevicesByInstance(self, device_class):
def getDeviceByID(self, id_num, hidden=False):

add device:
def addUdevLVDevice(self, info):
def addUdevDMDevice(self, info):
def addUdevMultiPathDevice(self, info):
def addUdevMDDevice(self, info):
def addUdevPartitionDevice(self, info, disk=None):
def addUdevDiskDevice(self, info):
def addUdevOpticalDevice(self, info):
def addUdevLoopDevice(self, info):
def addUdevDevice(self, info):

handle format:
def handleUdevDiskLabelFormat(self, info, device):
def handleUdevLUKSFormat(self, info, device):
def handleVgLvs(self, vg_device):
def handleUdevLVMPVFormat(self, info, device):
def handleUdevMDMemberFormat(self, info, device):
def handleUdevDMRaidMemberFormat(self, info, device):
def handleBTRFSFormat(self, info, device):
def handleUdevDeviceFormat(self, info, device):

actions:
def pruneActions(self):
def sortActions(self):
def processActions(self, dryRun=None):
def registerAction(self, action):
def cancelAction(self, action):
def findActions(self, device=None, type=None, object=None, path=None,

other:
def devices(self):
def filesystems(self):
def uuids(self):
def labels(self):
def leaves(self):

3 Do partition

Blivet use blivet.partitioning.doPartitioning() method to (re)configure all the partition requests so that each of them can get a proper location on the disks.

3.1 sort requests

3.2 remove new partitions

3.3 allocate partitions

3.4 grow partitions

3.5 manageSizeSet

4 Write changes to disks

5 Device factory