Ignite-UX and Mirrored Disks

May 14, 1999

Summary

There has been a great deal of misunderstanding around how the Ignite-UX tool set deals with mirrored disks. It is true that Ignite-UX does not currently have any built-in support for mirrored disks. Some have taken this to mean that Ignite-UX cannot be used on systems that have mirrored disks. **This is not the case. Ignite-UX is still a very valuable tool on systems that have mirrored disk configurations.**

This paper has two primary goals:
- ♦ Correctly and completely portray what Ignite-UX can and cannot do regarding mirrored disks.
- ♦ Explain how to use Ignite-UX on systems that have mirrored disks.

Today's State

Ignite-UX is a tool set used for the deployment (cold install) of HP-UX systems. Ignite-UX also includes the make_recovery tool, which is used to build a customized recovery tape to be used in the event of a root disk/volume group failure. Likewise, the make_net_recovery tool is used to build a network-accessible recovery image to be used in the event of a root disk/volume group failure.

MirrorDisk/UX is an extra-cost HP product. This product must be installed on the system to allow any disk mirroring capability.

None of the Ignite-UX tools are currently "aware" of mirrored disks.

Deployment

In the cold install (deployment) case this means:

- When describing how your system's disks will be configured via the Ignite-UX user interface, it is not possible to specify any disk mirroring.
- There are no Ignite-UX config file keywords that allow you to specify any disk mirroring.

It is, of course, quite possible to use Ignite-UX to cold install your system, and then set up your mirrored disks with the LVM commands or by using SAM. (Note that SAM cannot be used to set up mirrors for the root volume group.)

System Recovery

In the make_recovery and make_net_recovery cases, the system being recovered may have had mirrored disks before the system had the problem that forced the recovery. Of course, with a mirrored root disk, you're much less likely to ever need to recover a failed root disk via make_recovery or make_net_recovery. Nonetheless, mirroring your root disk does not cover all of the cases system recovery handles:

- There has been a catastrophic software error on the root disk that has also affected the mirror. For example, typing "rm -rf *" while in "/".
- The primary disk has a hardware failure and the mirror also fails before the primary can be replaced.
- You are about to update your system to a new operating system release or add new patches to the system. Creating a recovery configuration allows you to get back to a working system if the update does not succeed.

During such a recovery, Ignite-UX will not automatically re-create the mirrors. The mirrors need to be created via the LVM commands or SAM after the recovery is complete. (There is also a technique to hand-edit an Ignite-UX config file to make it re-build the mirrors as the last step in its process. This technique is described below).

How to Deploy Mirrored Disk Systems with Ignite-UX

When you're specifying how your disks should be configured in the Ignite-UX user interface, any disks that will eventually be mirrors should not be specified.

Once the cold install process has completed, you can set up your disk mirrors via the SAM command or by using the LVM commands directly. Note that

SAM cannot be used to mirror the root disk. Here's an example of how to mirror the root disk:

```
1. pvcreate -B /dev/rdsk/<second disk>
2. vgextend /dev/vg00 /dev/dsk/<second disk>
3. mkboot /dev/dsk/<second disk>
4. mkboot -a "hpux -lq" /dev/rdsk/<second disk>
5. lvextend -m 1 /dev/vg00/lvol1 /dev/dsk/<second disk>
6. lvlnboot -R
```

Repeat step 5 for any additional logical volumes that need to be mirrored.

How to Recover Mirrored Disk Systems with Ignite-UX via make_recovery

The make_recovery tool will create a recovery tape for a system with mirrored disks but it will not preserve the mirror configurations. After booting from the system recovery tape, disk mirrors will be lost and must be reconfigured using LVM commands.

There are basically two methods to do this. First, you can wait until the system is recovered and then execute the necessary LVM commands manually or use SAM. Second, it is possible to make some manual changes to the Ignite-UX configuration file created by make_recovery such that the LVM commands are executed as part of the recovery process itself. The LVM commands are the same in either case. This example will show how to integrate the commands into the Ignite-UX configuration file.

To integrate the configuration of disk mirrors into the recovery process, use make_recovery in preview mode. By invoking "make_recovery -p", the /var/opt/ignite/recovery/config.recover file will be created. Commands required to configure mirrored disks can be included as *post_config_cmds* and will be executed automatically after the system has been restored. After modifying /var/opt/ignite/recovery/config.recover, it's a good practice to run "instl_adm -T -f config.recover", to ensure that the syntax is correct.

Below is the portion of the configuration file that needs to be edited to restore disk mirrors after the recovery completes. The commands to reconfigure disk mirroring should be included in the configuration file as *post_config_cmds* (see instl_adm (4M)). This example shows how the root file system (lvol1) and primary swap (lvol2) can be mirrored to the disk at 0.5.0. The items that may need to be changed for your situation are in **bold font** in the example. The *post_load_cmd* and *final_cmd* clauses shown are part of the standard config.recover file created by make_recovery. They do not need to be altered by the user.

```
######## Clause produced by make_recovery - don't alter! ########
 post_load_cmd ="
    if [[ -f /tmp/install.vars ]]; then
```

```
          . /tmp/install.vars
    fi

    if [[ $RECOVERY_MODE = TRUE ]]; then
          cp /stand/vmunix /var/vmunix.makrec &&
          cp /var/vmunix.makrec /stand/vmunix
          rm -f /var/vmunix.makrec
    fi
 "

 ##############################################################
 ######## Begin user changes to add disk mirroring. ###########

 post_config_cmd +="
     # Recreate a mirror of the boot disk.

     # Make sure the disk is removed from the group by using
     # vgreduce. Alternatively you could edit the vg00
     # definition to remove the references to this disk.
     vgreduce /dev/vg00 /dev/dsk/c0t5d0

     # Make the disk contain a boot area.
     pvcreate -f -B /dev/rdsk/c0t5d0

     # Add the mirrored disk back to the group.
     vgextend /dev/vg00 /dev/dsk/c0t5d0

     # Copy the boot area to the disk.
     mkboot /dev/rdsk/c0t5d0

     # Turn off the quorum requirement
     mkboot -a \"hpux -lq\" /dev/rdsk/c0t5d0

     # Allocate the mirrors. Mirrors must be allocated for all
     # logical volumes that were previously mirrored. This
     # example illustrates primary swap and root.  You should
     # add others as needed. If /stand is in a separate volume, you
     # should do an lvextend for it first.
     lvextend -m 1 /dev/vg00/lvol1 /dev/dsk/c0t5d0
     lvextend -m 1 /dev/vg00/lvol2 /dev/dsk/c0t5d0

     # Update the BDRA and the LABEL file.
     lvlnboot -R
"

######## End user changes to add disk mirroring. ########
#########################################################


######## Clause produced by make_recovery - don't alter! ########
final_cmd="
    if [[ -f /tmp/install.vars ]]; then
        . /tmp/install.vars
    fi
    if [[ -a /var/tmp/remove ]]; then
        rm /var/tmp/remove
    fi
```

```
        if [[ -d /var/tmp/removedir ]]; then
            rmdir /var/tmp/removedir
        fi
        /usr/sbin/vgcfgbackup /dev/vg00
    "
```

After rebooting from the system recovery tape, invoke "vgdisplay -v", and/or "lvlnboot -v" to verify that disks have been mirrored as expected. The lifls (1M) command can also be used to ensure that the mirrored disk contains a LIF (Logical Interchange Format) image.

How to Recover Mirrored Disk Systems with Ignite-UX via make_net_recovery

The make_net_recovery tool will create a network recovery archive for a system with mirrored disks but it will not preserve the mirror configurations. After recovering from the network recovery archive, disk mirrors will be lost and must be reconfigured using LVM commands.

There are basically two methods to do this. First, you can wait until the system is recovered and then execute the necessary LVM commands manually or use SAM. Second, it is possible to manually create an Ignite-UX configuration file such that the LVM commands are executed as part of the recovery process itself. The LVM commands are the same in either case. This example will show how to integrate the commands into an Ignite-UX configuration file.

To integrate the configuration of disk mirrors into the recovery process, create a new Ignite-UX configuration file called /var/opt/ignite/clients/0x{LLA}/recovery/config.local. This config.local file will automatically be included into your recovery configuration for this client each time you run the make_net_recovery command. Note that make_net_recovery is run for you when you use the graphical user interface for network recovery.

If you already have recovery configurations for this client and would like them to use the config.local file, edit the /var/opt/ignite/clients/0x{LLA}/CINDEX file to include a reference to “recovery/config.local” in all of the configuration clauses.

Commands required to configure mirrored disks can be included as a *post_config_cmd* and will be executed automatically after the system has been restored. After creating this config.local file, it's a good practice to run "instl_adm -T -f config.local", to ensure that the syntax is correct.

Below is an example of a config.local file to restore disk mirrors after the recovery completes. This example shows how the root file system (lvol1) and primary swap (lvol2) can be mirrored to the disk at 0.5.0. The items that may need to be changed for your situation are in **bold font** in the example.

```
##############################################################
######## Begin user changes to add disk mirroring. ###########

post_config_cmd +="
    # Recreate a mirror of the boot disk.

    # Make sure the disk is removed from the group by using
    # vgreduce. Alternatively you could edit the vg00
    # definition to remove the references to this disk.
    vgreduce /dev/vg00 /dev/dsk/c0t5d0

    # Make the disk contain a boot area.
    pvcreate -f -B /dev/rdsk/c0t5d0

    # Add the mirrored disk back to the group.
    vgextend /dev/vg00 /dev/dsk/c0t5d0

    # Copy the boot area to the disk.
    mkboot /dev/rdsk/c0t5d0

    # Turn off the quorum requirement
    mkboot -a \"hpux -lq\" /dev/rdsk/c0t5d0

    # Allocate the mirrors. Mirrors must be allocated for all
    # logical volumes that were previously mirrored. This
    # example illustrates primary swap and root.  You should
    # add others as needed. If /stand is in a separate volume, you
    # should do an lvextend for it first.
    lvextend -m 1 /dev/vg00/lvol1 /dev/dsk/c0t5d0
    lvextend -m 1 /dev/vg00/lvol2 /dev/dsk/c0t5d0

    # Update the BDRA and the LABEL file.
    lvlnboot -R
"

######## End user changes to add disk mirroring. ########
#########################################################
```

After the system has been recovered, invoke "vgdisplay -v", and/or "lvlnboot -v" to verify that disks have been mirrored as expected. The lifls (1M) command can also be used to ensure that the mirrored disk contains a LIF (Logical Interchange Format) image.