Patching and Upgrading Tru64 UNIX and TruClusters

Martin Moore
Team Leader, Tru64 UNIX Support
HP Services
Session Topics

- Patching Tru64 UNIX and TruClusters
  - Background info
  - Best practices

- Upgrading Tru64 UNIX and TruClusters
  - Overview
  - Best practices

- New Hardware Delivery (NHD) kits
Types of patches

- Customer-Specific Patch (CSP)
  - Addresses a reported problem for a specific customer
    - Customized for customer’s system and patch level
    - Made available only to a specific customer
  - Unit testing performed
  - Shipped in *dupatch/setld* format
  - Rolled into a future release patch kit
Early Release Patch (ERP)

- A single patch available to all customers in response to a critical or widespread problem, such as:
  - Data integrity
  - Security
  - Detrimental to system operation
- Uses CSP packaging mechanism with slightly different documentation
- More extensive testing than CSP
- Available via patch web/FTP site
- Accompanied by Product Advisory and/or CERT advisory
- Rolled into a future release patch kit
Types of patches (cont.)

- **Release Patch Kit**
  - General distribution of patches
  - Available to all customers via FTP/web site:
  - Testing is a scaled version of a full functional release
  - Two types: Initial Patch Kit (**IPK**) or Aggregate Patch Kit
    - IPK contains fixes to prevent upgrade regression (forward porting of existing patches not in new version of O/S) plus fixes for new problems found in release qualification testing
    - Aggregate kits (subsequent base level kits after IPK) contain **cumulative** fixes for problems reported by customers or HP
    - Both may also include new functionality
Old Model:
- Manufactured and released serially
- Distributed on a 6-9 month cycle
- Order of priority dictated by Tru64 UNIX Support Engineering per customer demand
- Supported 3-7 patch kits per base level (delivered kits for all supported versions on all base levels)
  - 7 variants for BL16
  - 6 variants for BL17
  - 3 variants for BL18
Patch distribution model

- **New Model**
  - Ship 1-2 kits per base level
    - Deliver kits for releases with a high volume of incoming problem reports on a 90 day cycle
    - Deliver kits for releases with low volume less frequently
  - Deliver two patch kits simultaneously (or close to it)
  - Decrease the length of time between base levels

- **All release patch kits, aggregate and IPKs, go through an extensive test matrix before being released**
  - Scaled version of full quality testing that is performed in base OS/TruCluster development releases
Patch roadmap

- Recent and *forthcoming* patch kits
  - BL20: V5.1A pk3 (8/28/02), V5.1 pk6 (2/13/03)
  - BL21: V5.1B IPK (12/29/02), V5.1A pk4 (2/18/03)
  - BL22: V5.1B pk2 (5/6/03), V4.0G pk4, V4.0F pk8
  - BL23: V5.1A pk5

- Patch roadmap viewable at:
  
  http://www.tru64unix.compaq.com/docs/patch/roadmap.html
Supported versions

- Currently Supported Releases
  - V4.0F/TCR1.6 (Prior Version Support thru 30-Sep-2005)
  - V4.0G/TCR1.6 (PVS thru 30-Sep-2005)
  - V5.0A/TCR5.0A (PVS thru 30-Sep-2003)
  - V5.1/TCR5.1 (PVS thru 29-Feb-2004)
  - V5.1A/TCR5.1A (Standard support ends 29-Feb-2004; PVS from 1-Mar-2004 thru 28-Feb-2005)
  - V5.1B/TCR5.1B (Standard support through EOL)

http://www.compaq.com/services/software/ss_pvs_se_amap.html
Patch kit paradigm

- All patches are “a la carte”
  - Install/Remove all patches or selected patches (within limits; some patches depend on others)
  - **HP strongly recommends installing all patches**

- Patches can be installed as reversible or nonreversible
  - Reversible
    - Consumes additional disk space
    - Patches can be removed recursively
    - **Strongly recommended**
  - Nonreversible
    - Additional disk space not needed
    - Patches cannot be removed
Patch kit naming

- Release kits (as of BL18)
  - Example:
    
    OS Product|Version|Base Level|KitType|Kit#|MfgDate

Patch kit file T64V51B18AS0003-20011020.tar is read as:
- Tru64 UNIX (and TruCluster) software (T64)
- Version 5.1 (V51)
- Base level 18 (B18)
- Aggregate (selective installation) patch kit (AS)
- Patch Kit 3 (0003)
- Manufactured on October 20, 2001 (-20011020)
Patch kit naming (cont.)

- ERP’s and CSP’s – Existing style
  - Example: T64V51AB21-C0020100-12345-ES-20030611.tar
    - **T64V51AB21**: Product, version, base level (Tru64 UNIX V5.1A, BL 21)
    - **C0020100**: Patch number of one patch included in kit (patch 201.00) – kit may or may not include other patches
    - **12345**: Unique kit ID number (unique for this CSP)
    - **ES**: Kit type identifier
      - M = Manual patch (not installed with dupatch)
      - E = ERP (not present implies CSP)
      - S = Security patch
    - **20030611**: Manufacture date (June 11, 2003)
  - The kit number (12345 in this example) is the key field to uniquely identify this patch kit
Patch kit naming (cont.)

- ERP’s and CSP’s – New style (late 2003)
  - Example: T64KIT0012345-V51AB21-ES-20030611.tar
  - Minor changes from previous style
    - Individual patch number dropped
    - “KIT” and unique kit number moved to first field
  - Kit type identifier may include “C” to identify CSP
    - C and E mutually exclusive
Patch documentation

■ Publicly available at http://www.tru64unix.compaq.com/docs/patch

■ Patch Kit Installation Instructions
  - Installing and removing patches using the dupatch utility
  - Describes baselining techniques
  - Instructions for rolling and no-roll patching in clusters
  - Provides other information for working with patches

■ Patch Summary and Release Notes
  - Provides information about specific patches
  - Summarizes Tru64 UNIX operating system patches included in the kit
  - Summarizes TruCluster patches included in the kit
Best practices for patches

- Patch “Best Practice” document on-line at http://www.tru64unix.compaq.com/docs/best_practices/
  - Not the same as this presentation

- Back up your system first
  - If catastrophe happens while patches are installing, you may have to restore system (or undo rolling upgrade)
  - At a minimum: vdump the /, /usr, and /var filesystems

- Review patch kit release notes
  - Especially if you have CSP’s or ERP’s installed
  - If questions remain about whether a CSP or ERP fix is included in a release patch kit, contact HP support
Best practices for patches (cont.)

- Check for presence of CSP’s / ERP’s
  - “Show System Patch History” option from “Patch Tracking” menu
  - (or) `dupatch -track -type kit` command

- Remove CSP’s/ERP’s installed by dupatch
  - Dupatch doesn’t know if fixes are in the release kit, so it won’t install the kit on top of them
  - Forthcoming: dupatch will be able to reconcile CSP’s with release kits and install kits over CSP’s when fixes are included in the release kits (late 2003)
Best practices for patches (cont.)

- Run “Baseline Analysis/Adjustment” in dupatch
  - Checks for additional missing or inconsistent files that block the installation of specific patches
  - Could be manual (non-dupatch) patches, third-party or freeware utilities, locally modified programs, etc.
  - Selectively enable patch installation if you determine that’s the right course (or don’t enable if it’s not)

- Put a meaningful comment in installation log
  - e.g., “Installing security ERP for CERT advisory XXXX”
  - Dupatch transactions logged in /var/adm/patch/log

- Always install patches reversibly
  - Preserve backed-up files in case you need to reverse patches
    - Location selectable; /var/adm/patch/backup by default
Patching clusters

- **Rolling Patch**
  - Allows cluster to remain up while patch operations are performed sequentially on cluster members
  - Maintains availability of critical applications
  - Can combine aggregate kit + ERP’s/CSP’s in a single roll
  - Can combine installupdate (and/or NHD) + patches in a single roll

- **No-Roll Patch**
  - Introduced in BL19 patch kits (Spring 2002)
  - Allows patches to be managed on the whole cluster at once rather than using rolling patch
  - Provides the ability to patch a cluster quickly with minimal downtime
Version switch technology

- Implemented in V5.1 pk4 and V5.1A IPK (Fall 2001)
- Applicable only in cluster environment
- Prevents the introduction of potentially incompatible new cluster features until all members have been updated with necessary components
- `clu_upgrade --switch` command activates new version
- Use of the version switch requires a cluster-wide shutdown to remove those patches requiring the version switch
- `clone_versw_undo` script for V5.1
- `evm_versw_undo` script for V5.1A and V5.1B
Phases of the rolling upgrade process

- **Set up**
  - Check for existing roll
  - Ask if OS update or patch
  - Reboot Other Members
  - Create Tagged Files

- **Preinstall**
  - Check state of members
  - Check Tagged Files

- **Install**
  - Install/remove patch(es) using `dupatch` utility
  - Verify member installed

- **PostInstall**
  - If Update, checks version
  - Perform Analysis
  - Setup scripts

- **Roll**
  - Reboot
  - Switch

- **Switch**
  - Switch version
  - Ensure all members rolled

- **Clean**
  - Delete Tagged Files
Rolling upgrade: Preparation

- Back up the cluster
- Upgrade firmware if necessary
- Choose a lead member
- `clu_upgrade -v check_setup <lead_member_id>`
clu_upgrade setup

- Verifies:
  - System is part of a cluster configuration
  - No roll is currently in progress
- Asks whether OS upgrade, NHD, or patch (or combo)
- Creates tagged files
  - For best performance, relocate CFS server for /, /usr, /var to the member running this step
- Instructs user to reboot all cluster members except the lead member
  - Non-lead members now running on tagged files
  - Usually done one member at a time to maintain quorum
- Usually takes 45 minutes to 2 hours
clu_upgrade preinstall

- Executed on the lead member
- Verifies:
  - Tagged files are present
  - Tagged files match the inventory
  - All non-lead cluster members are running on tagged files
- Usually takes 15 to 30 minutes
clu_upgrade install

- Executed on the lead member
- Patches installed/removed using `dupatch` utility
- At this point, only the lead member is running with patches installed (or removed); all other members are running with tagged files and patches uninstalled (or not yet deleted)
- `dupatch` can be run multiple times
  - Can install from multiple patch kits (aggregate and/or ERP and/or CSP)
  - Multiple patch deletions and/or installations can occur within the same roll
clu_upgrade postinstall

- Executed on the lead member only
- Checks to see if dupatch has been executed before this step is completed
**clu_upgrade roll**

- All non-lead members must be rolled
- This is an iterative step
  - Repeated on each member until complete
  - Members can be rolled in parallel if cluster quorum is maintained
- Only member-specific files are modified because the shared files are already in place from the roll of the lead member
- Sets up *it(8)* job that will be executed on reboot to perform the roll
clu_upgrade roll (cont.)

- Runs in single-user mode on each member
- Backs up member’s member-specific files
- Reboots the member and upon reboot:
  - Copies and configures member-specific files from member0
  - Builds new custom kernel for each member
  - Reboots customized kernel
- All cluster members now have patches configured
clu_upgrade switch

- Run once after all members have completed their roll
- Any installed patch controlled by a version switch will be enabled on all cluster members
- Prior to this phase, a patch containing a version switch can be deleted
- After this step, the roll itself must be completed; the user is no longer permitted to undo steps
After the version switch has been thrown, removing a patch containing a version switch must follow a special procedure:

1) Complete the roll if still active
2) Run the associated undo script (supplied with the patch subset):
    • `clone_versw_undo` (V5.1)
    • `evm_versw_undo` (V5.1A and up)
    • Allows the patch containing a version switch to be removable
3) Remove the patch
clu_upgrade clean

- Verifies that the switch stage has completed
- Removes the tagged files
- Ensures proper cleanup so that future rolls can be performed
- Creates an archive of the logs and status files under /cluster/admin/clu_upgrade/history
- Usually takes 30 to 90 minutes
  - As with other steps, run on member that is CFS serving /, /usr, and /var filesystems (or relocate them)
No-roll patch process

- Introduced in BL19 (V5.1A PK2/V5.1 PK5) as an alternative to the rolling patch process
  - Some users considered the cluster to be unavailable while in the midst of a roll
  - User may be applying only a single patch (ERP or CSP) and doesn’t need or want to use the lengthy roll process
- Provides the user with a procedure to maintain patches with minimal downtime and less manual intervention
- Applies patches to all members automatically
No-roll patch process (cont.)

- Requires less disk space than rolling patch process (no tagged files)
- Cluster services are not available during this process
- Cluster restarted after all members have been patched
No-roll patch – Preconditions

- Quorum configuration is important
  - Quorum must be configured such that all cluster members can go down (serially or in tandem) while maintaining quorum
  - Detailed information provided in the *Cluster Installation Guide*

- Cluster members must be in multi-user mode because EVM is used to generate cluster-wide events
No-roll patch timeline

Set-up
- Install - 1st member
- Patch Selection
- Analysis
- Ask if no-roll is desired
- Check for cluster & not in a roll

Install - 1st member
- Install Patches to 1st member
- Rebuild kernel if necessary

Install - all other members
- Apply member specific files
- Rebuild kernels if necessary
- Reboot all cluster members

PostInstall
- Reboot all cluster members

Switch
- Version Switch (if needed)
No-roll patch - Set-up

- In *dupatch*, user selects install/delete patches
- Checks for correct state:
  - Configuration is a cluster
  - A cluster roll is not in progress
- Confirms that a no-roll operation is desired
- Verifies that cluster is in multi-user mode
- Performs analysis
  - Disk space
  - Dependencies
- User selects the list of patches to be installed
- Remainder of the no-roll process is automated
No-roll patch timeline

Check for cluster & not in a roll

<table>
<thead>
<tr>
<th>Setup</th>
<th>Install - 1st member</th>
<th>Install - all other members</th>
<th>PostInstall</th>
<th>Switch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ask if no-roll is desired</td>
<td>Patch Selection</td>
<td>Install Patches to 1st member</td>
<td>Rebuild kernel if necessary</td>
<td>Reboot all cluster members</td>
</tr>
<tr>
<td>Analysis</td>
<td>Cluster down to run level 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Apply member specific files</td>
<td>Rebuild kernels if necessary</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
No-roll patch - Install first member

- “First member” is the cluster member running dupatch
- Cluster is brought down to run level 2 via a cluster-wide event
- Patches are loaded and configured on the first cluster member
- If any patches require a kernel build, a new kernel is built automatically
No-roll patch timeline

Setup

- Check for cluster & not in a roll
- Ask if no-roll is desired
- Cluster down to run level 2
- Analysis
- Patch Selection

Install - 1st member

- Install Patches to 1st member
- Rebuild kernel if necessary

Install - all other members

- Rebuild kernels if necessary
- Apply member specific files

PostInstall

- Reboot all cluster members

Switch

- Version Switch (if needed)
No-roll patch - Install all members

- A second cluster-wide event is issued on all running cluster members:
  - Executes `/var/adm/patch/patch_member_noroll` script
  - Operation is done in parallel to minimize downtime

- For a down member:
  - Posts an `it(8)` job which, upon reboot, will run
    `/var/adm/patch/patch_member_noroll`
  - Populates member-specific files and automatically rebuilds kernel, if necessary
No-roll patch timeline

Setup
- Check for cluster & not in a roll
- Ask if no-roll is desired
- Patch Selection
  - Analysis
- Cluster down to run level 2

Install - 1st member
- Install Patches to 1st member
- Rebuild kernel if necessary

Install - all other members
- Rebuild kernels if necessary
- Apply member specific files

PostInstall
- Reboot all cluster members

Switch
- Version Switch (if needed)
No-roll patch - Postinstall

- Status of all members is checked
- Cluster-wide event is issued that causes all members in the cluster to be rebooted
- Cluster members will reboot in a fashion that will not lose cluster quorum during the reboot
No-roll patch timeline

Set-up
- Check for cluster & not in a roll
- Ask if no-roll is desired
- Analysis
- Patch Selection

Install - 1st member
- Cluster down to run level 2
- Install Patches to 1st member
- Rebuild kernel if necessary

Install - all other members
- Apply member specific files
- Rebuild kernels if necessary

PostInstall
- Reboot all cluster members
- Version Switch (if needed)

Switch
If a patch containing a version switch has been installed, the user must manually throw the version switch for the cluster as a whole by running the script /var/adm/patch/noroll/noroll_versw

All cluster members must be up at this time

Once the switch has been thrown, if the version switch patch needs to be removed, the undo script must be run followed by patch deletion (identical to rolling patch process)

It is recommended that the cluster be rebooted after throwing the switch
Handling errors in no-roll patch

- At the start of a no-roll operation, *dupatch* determines which members are presently up
  - Members that were known to be running at the start of the no-roll operation, but do not respond to later cluster events, will cause the no-roll operation to be suspended until the problem has been corrected by the user

- Status from the script is reported in a file in the shared directory `/var/adm/patch/noroll_results`
## Rolling vs. no-roll comparison

<table>
<thead>
<tr>
<th></th>
<th>Rolling Patch</th>
<th>No-Roll Patch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster Services</td>
<td>Available</td>
<td>Unavailable</td>
</tr>
<tr>
<td>Time to Complete</td>
<td>Lengthy</td>
<td>Minimal</td>
</tr>
<tr>
<td>Reboots per Member</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>Additional Disk Space Needed</td>
<td>Yes (for tagged files)</td>
<td>No</td>
</tr>
<tr>
<td>Automated Process</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Combine Patch with OS Upgrade</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>
Upgrading Tru64 UNIX

installupdate utility does the following:
  - Checks for file type conflicts
  - Checks for sufficient disk space
    • Provides several options to free up space
  - Notifies of possibly conflicting layered products
  - Optionally removes blocking layered products
  - Updates base OS, TCR, and WLS subsets to new version
<table>
<thead>
<tr>
<th>To get to this version:</th>
<th>You can do one update from:</th>
</tr>
</thead>
<tbody>
<tr>
<td>V4.0D</td>
<td>V4.0A, V4.0B, V4.0C</td>
</tr>
<tr>
<td>V4.0E</td>
<td>V4.0B, V4.0C, V4.0D</td>
</tr>
<tr>
<td>V4.0F</td>
<td>V4.0D, V4.0E</td>
</tr>
<tr>
<td>V4.0G</td>
<td>V4.0D, V4.0E, V4.0F</td>
</tr>
<tr>
<td>V5.0</td>
<td>V4.0D, V4.0F</td>
</tr>
<tr>
<td>V5.0A</td>
<td>V4.0F, V5.0</td>
</tr>
<tr>
<td>V5.1</td>
<td>V4.0G, V5.0A</td>
</tr>
<tr>
<td>V5.1A</td>
<td>V5.0A, V5.1</td>
</tr>
<tr>
<td>V5.1B</td>
<td>V5.1, V5.1A</td>
</tr>
</tbody>
</table>
TruCluster upgrade paths

- A rolling upgrade can only be performed to the next higher version of TCR software
  - V1.4A to V1.5
  - V1.5 to V1.6
  - V5.0A to V5.1
  - V5.1 to V5.1A
  - V5.1A to V5.1B

- No rolling upgrade path from TCR 1.6 to TCR version 5 due to introduction of Single System Image

- So clusters must roll in small steps
  - V5.1 cluster must roll to V5.1A before rolling to V5.1B
  - Standalone V5.1 system could go to V5.1B in one step
Best practices for upgrades

- Back up your system first
  - If catastrophe happens while subsets are loading, you’ll probably have to restore system (or undo rolling upgrade)
  - At a minimum: vdump the /, /usr, and /var filesystems

- Check release notes and installation guide
  - Is platform still supported? (Very old ones may not be)

- Upgrade firmware if needed
  - After upgrade, re-run ECU on systems with EISA buses

- Verify AdvFS domains

- Run `cdslinvchk` to verify CDSL integrity

- No need to install patches on old version prior to upgrade (but do install them on the new version)
Things to consider

- How many upgrades to do vs. a full install?
  - Time/risk tradeoff vs. effort to recustomize system

- 4.0F to 5.0A upgrade is problematical
  - Avoid if possible by choosing a different path: e.g., 4.0F to 4.0G to 5.1 to 5.1B instead of 4.0F to 5.0A to 5.1A to 5.1B

- If root directory is mirrored via LSM, consider unencapsulating it during upgrade
  - One less layer of complexity
  - Former mirror plex is a snapshot of pre-upgrade root
New Hardware Delivery (NHD)

- Periodic kits issued to provide Tru64 UNIX support for new hardware options
  - Six kits issued to date; more to come
  - Provides support for V5.1A and V5.1B

- NHD-6 added support for Broadcom 5703 GbE NIC, SmartArray RAID controller, DS20L, and additional 5.1B functionality for EV7 systems (ES47, ES80, GS1280)

- Where to get NHD kits?
  - Factory installed on new systems
  - CD-ROM bundled with Tru64 UNIX media kit or available separately (part number QA-MT4AX-H8)
  - http://ftp.support.compaq.com/public/unix/v5.1a/nhd/ (or “v5.1b” instead of “v5.1a”)
Interex, Encompass and HP bring you a powerful new HP World.