ServiceControl Manager Tools

Donald Suit

Hewlett Packard

3404 Harmony Road, MS 99

Fort Collins, CO 80528-9599

(970) 898-0327

(970) 898-2838 fax

dsuit@fc.hp.com

ServiceControl Manager Tools

- Introduction
- Purpose
- SCM Overview
- SCM Tool Details

Introduction

- Manage a cluster of nodes from a Central Management Station(CMS)
- Manage a set of configuration tools on the CMS
- Execute the tools from the CMS

Purpose

- Provide more controlled mechanism than remsh
- Build toolkit of favorite scripts and routine tasks for node distribution
- Create tools for trouble–shooting across a set of managed nodes
- Use convenience of HP–provided tools
  - Ignite-UX (IUX)
  - Software Distributor (SD)

SCM Overview

- Users
- Managed Nodes
- Node Groups
- Roles
- Authorizations
- Tools
- Logs
- Repository

SCM User

- Identifies users who have access to the SCM using system names and Ids
- Defines SCM privileges for users
  - Trusted
  - Non-trusted
- Provides access to additional user information, gecos data

SCM Managed Node

- Identifies nodes that are managed by the SCM
- Provides additional information, e.g. IP addresses

SCM Node Group

- Provides capability to logically associate clusters of managed nodes
- Allows nodes to be associated to multiple groups

SCM Role

- Names a function that can be associated with nodes, users, and tools

- Provides 16 available roles

- Assigns one role as "Master Role" that is immutable

- Allows SCM administrator to modify names of other roles

SCM Authorization

- Associates a user, role, and node
- Defines which user can execute which tool on which node

SCM Tool

- Defines an activity that can be executed across a set of managed nodes by an authorized user
- Provides two types of activities
  - Allows file copying
  - Allows command execution
- Permits both activities in one tool, file copying first then command execution

SCM Log

- Captures activities performed in the SCM
- Identifies types of activities and results

SCM Repository

- Provides data persistence
- Stores all types of SCM data
- Implemented with the NDS mechanism

SCM Tool Details

- Tool Attributes
- Managing Tools
- Executing Tools

SCM Tool Attributes

- Name
  - Identifies tool for addition, modification, removal, listing, or execution.
  - Required
- Category
  - Associates tool logically with other tools.
  - Defaults to "Local Tools"

SCM Tool Attributes (cont.)

- Description
  - Describes tool
  - Optional

- Owner
  - Defines the SCM user that "owns" the tool
  - Optional

SCM Tool Attributes (cont.)

- Owner (cont.)
  - If specified
    - Only "Master Role" role valid, disables other roles.
    - "Owner" can modify tool but can not modify roles or owner attribute.
  - If not specified
    - enables all roles
    - only SCM administrator can modify tool.

SCM Tool Attributes (cont.)

- Comment
  - May be multi-line text field providing instructions, comments, warnings, etc.
  - Optional
- Command
  - Complete command line if no arguments need to be entered at run time; otherwise, initial invariant portion of command line to execute.
  - Required if no file-copy pair specified

SCM Tool Attributes (cont.)

- Parameters
  - List of argument entries
  - Optional
  - Composed of
    - Prefix
    - Prompt
    - Required flag

SCM Tool Attributes (cont.)

- Parameters (cont.)
  - Prefix
    - Appended to command string while building command for execution.
    - Optional
  - Prompt
    - Describes value concatenated to command string
    - Used by SCM GUI
    - Optional

SCM Tool Attributes (cont.)

- Parameters (cont.)
  - Required flag
    - Indicates if argument is required
    - Required
  - Either prefix or prompt must be defined.

SCM Tool Attributes (cont.)

- File–copy pairs
  - Lists files to be copied.
  - Consists of
    - Source pathname on the CMS
    - Destination pathname on the managed node
  - SCM prohibits more than one file–copy pair with same destination in one tool
  - Required if no command specified

SCM Tool Attributes (cont.)

- Execution user
  - Defines UID under which tool executes on a managed node
  - Optional, if not specified, SCM executes tool using UID of user who initiated tool execution

SCM Tool Attributes (cont.)

- Default targets
  - Specifies default target managed nodes if user does not specify at run time.
  - Possible values
    - None specified, i.e. no defaults
    - "ALL" - all authorized managed nodes
    - "CMS" - run on managed node specified as CMS
    - Specific node (user must be authorized)
  - Optional

SCM Tool Attributes (cont.)

- Log flag
  - Indicates whether to save stdout and stderr results from tool execution of tool to log
  - Be default, results are saved

- Launch–only flag
  - Indicates whether Distributed Task Facility(DTF) agent should wait for completion of command and capture results
  - By default, agent waits for command completion.

Managing SCM Tools

- Defining tools
- Adding tools
- Modifying tools
- Removing tools
- Displaying tool information

Defining Tools

- SCM provides two tool definition mechanisms
  - GUI windows
  - CLI tool definition file
- Any SCM user may define attributes except owner and roles
- Only SCM administrators may modify owner and roles

Defining Tools (cont.)

- Sample GUI tool properties screen

Defining Tools (cont.)

- Sample tool definition

```
SSA tool "Sample Tool" {
    description "This is a one line description of a sample tool"
    comment "This is a longer description to provide more information,
            perhaps even usage information"
    revision "1.0"
    owner dsuit
    category "Don's Tools"
    copy {
        /var/etc/sourcefile : /var/etc/destinationfile,
        /var/etc/source2    : /var/etc/dest2
    }
    execute {
        command "ls "
        arguments {
            "-l ": ,
            :"Enter path: " optional
        }
        nolaunch
        log
        user root
    }
}
```

Adding Tools

- Non-SCM administrator
  - May define all attributes but owner and roles
  - SCM sets owner field to user's UID
  - "Master Role" only assigned role
- SCM administrator
  - May define all attributes, including owner and roles
- User cannot "add" existing tool

Adding Tools (cont.)

- Sample GUI window

Adding Tools (cont.)

- Sample CLI

```
mxtool -a -f mytools
```

Modifying Tools

- Users may only modify existing tools
- Non-SCM administrators users
  - Can only modify "owned" tools
  - Can not modify owner or roles
  - May require intervention from SCM administrator to set owner field if owner cleared
- SCM administrator
  - Can modify any tool
  - Can modify owner and roles

Modifying Tools (cont.)

- Sample GUI window

Modifying Tools (cont.)

- Sample CLI

```
mxtool -m -f mytools
```

Removing Tools

- Only SCM administrators may remove tools

Removing Tools (cont.)

- Sample GUI window

Removing Tools (cont.)

- Sample CLI

```
mxtool -r -t "Sample Tool"
```

Displaying Tool Information

- GUI
  - Provides hierarchical display based on category
  - Displays tools accessible by user

- CLI
  - Lists tool names
  - Provides tabular listing of category, name, description
  - Provides screen-viewable detailed listing
  - Provides definition file formatted output

Displaying Tool Information (cont.)

- Sample GUI window

Displaying Tool Information (cont.)

- CLI Samples

```
#Simple list command of all tools using no option
mxtool
#Simple list command of all tools using name option
#Same as not specifying any options
Mxtool -ln
#Tabular list command of all tools
mxtool -lt
#Tabular list command of one category
mxtool -lt -c "Local Tools"
#Detailed list command
mxtool -ld
#Definition file formatted command of one tool definition
mxtool -lf -t "Sample Tool" > mySampleTool
```

Executing Tools

- Generating executable command
- Determining authorized nodes
- Running executable command
- Canceling and killing tool execution
- Examining execution results

Generating executable command

- Requires user to
  - Specify tool name
  - Specify nodes if no defaults specified or desired
  - Specify argument values if needed
- Constructs complete command line
  - Takes command from definition
  - Concatenates prefixes from definition
  - Adds user–specified argument values

Generating executable command (cont.)

- Sample GUI Window

Generating executable command (cont.)

- Sample CLI

```
mxexec -t "My Tool Name"
mxexec -t "My Tool Name" -A value1 "value 2" "value 3"
mxexec -t "My Tool Name" -n target1 target2 target 3
mxexec -t "My Tool Name" -O outputfilepath
mxexec -t "My Tool Name" -o outputDirectoryPath
```

Determining authorized nodes

- Generates target list
  - Specified nodes override default targets
  - If no specified nodes
    - If "CMS" specified as default, CMS node is target
    - If "ALL" specified as default,
      - Iterates through all managed nodes and generates authorized target list
    - If one node specified as default, uses that node

Determining authorized nodes (cont.)

- Determines if targets are authorized nodes
  - All targets must be authorized
  - Execution user must be authorized tool's roles on every target node
- If all targets authorized, passes command line, file information, target list to DTF daemon

Running Executable Command

- DTF daemon manages executable commands as tasks on CMS

- DTF daemon transitions tasks to following states:
  - Pending
  - Contacting target
  - Copying files
  - Running tool
  - Complete

Running Executable Command (cont.)

- DTF agent manages task on target
- DTF agent writes files and sets files' ownership and permissions
- DTF agent forks a process to run the command, establishes clean process environment, and executes POSIX shell with command line as argument.

Canceling and killing tool execution

- SCM users may cancel or kill tool execution
- When canceling a task, DTF daemon:
  - Interrupts file copying
  - Does not transmit command to DTF agent
- If task is in running state, cancel is ineffective
- When killing a task, DTF daemon sends kill requests to each agent

Examining execution results

- If tool not launch-only, task results sent user at task completion

- If task output still managed by DTF, user may request output using task id

- If task output sent to log file, user may examine log file

Summary

- SCM tools provide mechanism for configuring and managing HP-UX clusters
- SCM provides GUI and CLI access to define and execute tools
- SCM provides access to and logging of tool execution results