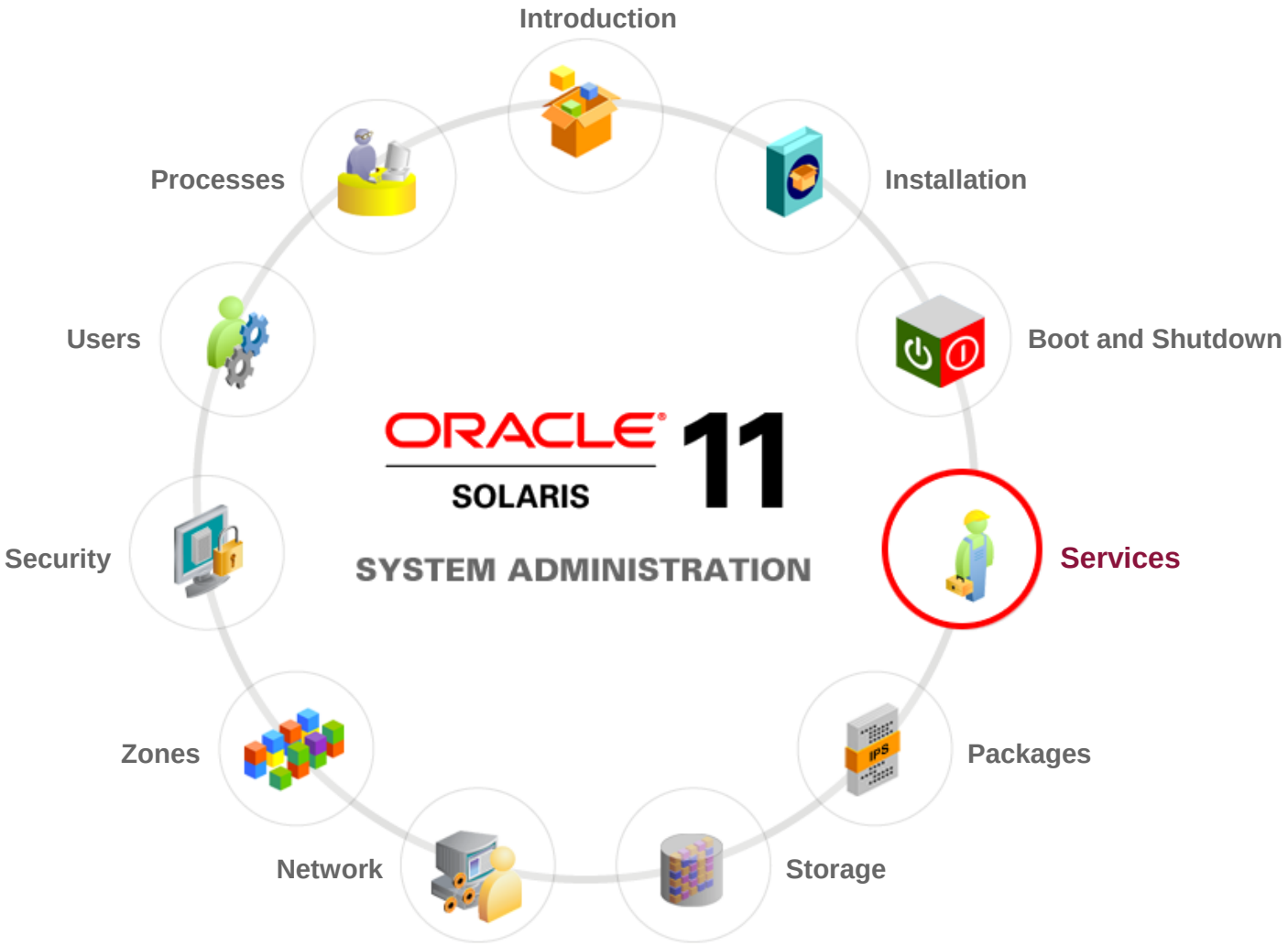
4

Administering Services by Using SMF

Workflow Orientation



Objectives

After completing this lesson, you should be able to:

- Explain the SMF feature and its components
- Administer SMF services

Importance of Services Administration

Services administration is required to ensure that:

- System and application services run smoothly and efficiently
- Systems continue to be available by providing all essential services even during a system failure

Service Management Facility

- Provides a framework for managing the following:
  - System and application services
  - Interaction of services with other dependent services
- Contains information about:
  - Procedures to start, stop, and restart services
  - Service startup behavior and status
  - Misconfigured services (such as an explanation of why a service is not running)
- Provides an individual log file for each service

SMF Capabilities

- Booting faster
- Restarting failed services
- Inspecting services
- Managing services
- Configuring services
- Auditing service changes
- Securely delegating tasks
- Creating new services
- Debugging service problems
- Configuring failure notification
- Converting `inetd.conf` configurations to SMF services
- Converting SMF service properties to configuration files

SMF Service

- An entity that provides a resource or list of capabilities to applications and other resources
- The software state of a device (for example, a configured network device or mounted file system)
- Structured within SMF by:
  - Category (examples: `application`, `network`, `system`)
  - Service name (examples: `login`, `SSH server`, `hostid`)
  - Instance name: Specific configuration of a service (example: `default`)

Note: There can be multiple instances of a service.

Service Instance

Example FMRI:

`svc:/system/filesystem/root:default`

where:

- The prefix `svc` indicates that this service is managed by SMF
- The highest category of the service points to the system facilities (`system`)
- Within `system`, the next level category is `filesystem`
- The next lower category is `root`, which points to the `root` file system
- The service name is `system/filesystem/root:default`
- An instance of the service is `default`

Service Models

SMF services are one of the following three models:

SMF Model	Description
Transient service	The service performs some task, and then exits without starting any long running processes.
Child or wait service	The service is restarted whenever its child process exits cleanly. A child process that exits cleanly is not treated as an error.
Contract or daemon service	The service starts a long-running daemon or starts several related processes that are tied together as part of a *service contract*. The contract service manages the processes that it starts and any dependent services and their start order. You only need to manage the high-level service.

Service States

A service instance can have different states, as listed in the following table:

Service State	Description
`online`	Enabled and successfully started
`offline`	Enabled but not yet running or available to run
`offline*`	Process that is in the state of starting
`disabled`	Not enabled and not running
`legacy_run`	Running. The service is not directly managed by SMF, but it was started at some point.
`uninitialized`	Starting up. This state is the initial state for all services before their configuration has been read.

Service States

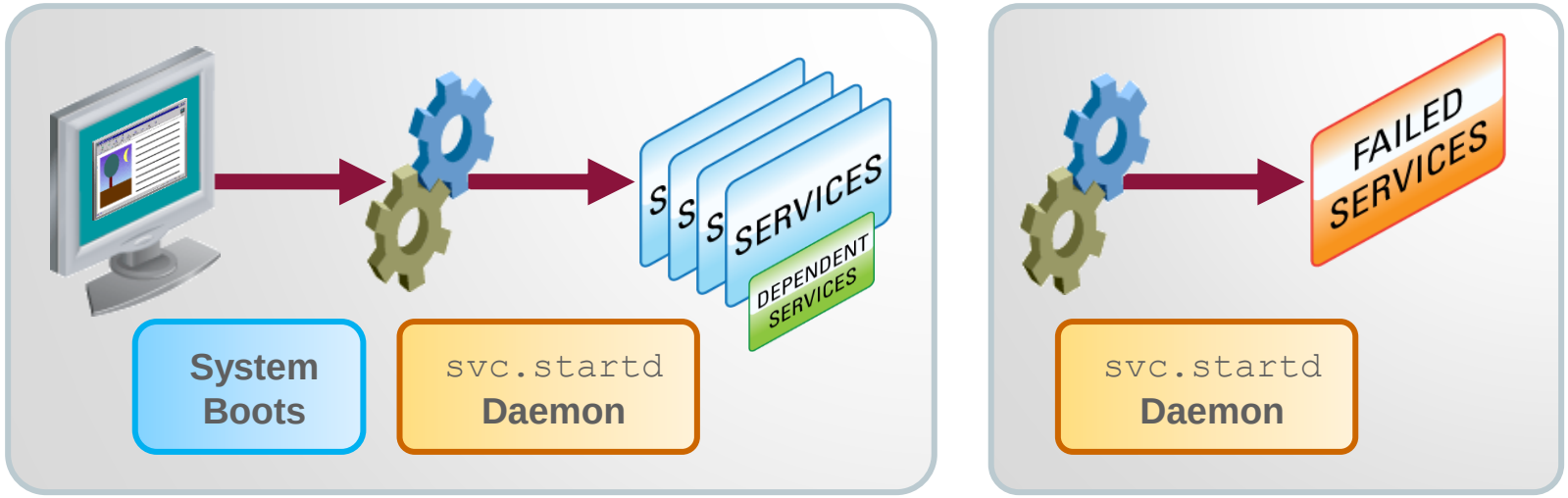
Service State	Description
`maintenance`	Error encountered that requires administrative intervention
`degraded`	Enabled but running at limited capacity

Service Configuration Repository

- Stores state and configuration information about each service instance
- Is named `/etc/svc/repository.db`
- Is managed by the `svc.configd` daemon

SMF Master Restarter Daemon (`svc.startd`)

- Manages service dependencies for the entire system
- Makes sure that the system boots properly
- Is responsible for starting services, restarting services, and shutting down services



Quiz

What is the service category in the `svc:/network/ssh:default` service FMRI?

a. `svc`

b. `network`

c. `ssh`

d. `default`

Quiz

What is the service category in the `svc:/network/ssh:default` service FMRI?

a. `svc`

b. `network`

c. `ssh`

d. `default`

Quiz

Which of the following daemons is responsible for starting services?

a. svc.startd

b. /etc/init

c. svc.configd

d. repository.db

Quiz

Which of the following daemons is responsible for starting services?

a. svc.startd
b. /etc/init
c. svc.configd
d. repository.db

Quiz

If a service instance is in the state of starting, which of the following states will it be identified with?

a. uninitialized
b. online
c. offline
d. offline*

Quiz

If a service instance is in the state of starting, which of the following states will it be identified with?

a. uninitialized
b. online
c. offline
d. offline*

Agenda

- Describing SMF and Its Components
- **Administering SMF Services**

Administering SMF Services

- Monitoring services with `svcs`
  - Listing services information
  - Displaying the status of a service
  - Displaying the service dependents
  - Displaying the dependencies of a service
- Administering services with `svcadm`
  - Disabling a service
  - Enabling a service
  - Restarting a service
- Setting up service state transition notifications
  - Installing the `smtp-notify` package
  - Enabling the `smtp-notify:default` service
  - Configuring service state transition notifications
    - Service state transition notification: Example
  - Managing service state transition notifications

Listing Services Information

To list all the services currently running in the system, run the `svcs` command.

```
# svcs
STATE          STIME    FMRI
legacy_run     1:25:08 lrc:/etc/rc2_d/S47pppd
legacy.run     1:25:08 lrc:/etc/rc2_d/S81dodatadm_udaplt
legacy_run     1:25:08 lrc:/etc/rc2_d/S89PRESERVE
<output truncated>
```

To list all the services defined in the system, run the `svcs -a` command.

```
# svcs -a
STATE          STIME    FMRI
legacy_run     1:25:08 lrc:/etc/rc2_d/S47pppd
legacy.run     1:25:08 lrc:/etc/rc2_d/S81dodatadm_udaplt
legacy_run     1:25:08 lrc:/etc/rc2_d/S89PRESERVE
disabled       1:23:38 svc:/system/device/mpxio-upgrade:default
<output truncated>
```

Displaying the Status of a Service Instance

To display the status of a service, run the `svcs -l` *FMRI* command.

```
# svcs -l svc:/network/ssh:default
fmri         svc:/network/ssh:default
name         SSH server
enabled      true
state        online
next_state   none
state_time   August 13, 2015 08:37:35 AM IST
logfile      /var/svc/log/network-ssh:default.log
restarter    svc:/system/svc/restarter:default
contract_id  122
manifest     /etc/svc/profile/generic.xml
manifest     /lib/svc/manifest/network/ssh.xml
dependency   require_all/none svc:/system/filesystem/local (online)
dependency   optional_all/none svc:/system/filesystem/autofs (online)
dependency   require_all/none svc:/network/loopback (online)
dependency   require_all/none svc:/network/physical:default (online)
dependency   require_all/none svc:/system/utmp (online)
<output truncated>
```

Dependency Groupings

Each dependency is assigned to one of the following groupings. The grouping defines how dependencies in that grouping are satisfied.

- `require_all`
- `require_any`
- `optional_all`
- `exclude_all`

Dependency Groupings

Mode	Descriptions
require_all	This dependency is satisfied when **both** of the following conditions are met: • All the service dependencies in this grouping are running, either online or degraded. • All the file dependencies in this grouping are present.
require_any	This dependency is satisfied when **either** of the following conditions is met: • At least one of the service dependencies in this grouping is running, either online or degraded. • At least one of the file dependencies in this grouping is present.
optional_all	This dependency is satisfied when **all** the service dependencies in this group meet **either** of the following conditions: • The service is running, either online or degraded. • The service requires administrative action to run. The service is not present, is incomplete, is disabled, is in maintenance, or is offline waiting for dependencies that require administrative action to start.
exclude_all	This dependency is satisfied when **both** of the following conditions are met: • All the service dependencies in this grouping are disabled, in maintenance, or not present. • All the file dependencies in this grouping are not present.

Determining Whether a Service Will Automatically Restart

The value of the `restart_on` attribute will determine whether a service will restart after a dependency stop or refresh event.

`require_all`, `require_any`, or `optional_all` Dependency	**Value of Dependency `restart_on` Attribute**			
Stop or Refresh Event	**`none`**	**`error`**	**`restart`**	**`refresh`**
Stop due to error	No restart	Restart	No restart	Restart
Other stop	No restart	No restart	No restart	Restart
Refresh	No restart	No restart	Restart	Restart

Viewing Service Log Files

The `svcs` command with the `-L`, `-l`, or `-x` option shows the full path name of the log file for the specified service instance.

- To view the last few lines of a log file, run the `svcs -xL` command.
- To view the complete log file, run the `svcs -Lv` command.
- To view the log file in an editor or view the last *n* entries, operate on the output of the `svcs -L` command.

Contract Services

The service starts a long running daemon or starts several related processes that are tied as part of a *service contract*. The contract service manages the processes and services it starts. You only need to manage the high-level service.

To view the process IDs and command names of the processes started by a contract service instance, run the `svcs -p` command.

```
$ svcs -p net-snmp
STATE    STIME    FMRI
online   17:57:26 svc:/application/management/net-snmp:default
         17:57:26         5022 snmpd
```

Displaying the Service Dependents

To display service dependents, run the `svcs -D` *`FMRI`* command.

```
# svcs -D svc:/network/ssh:default
STATE          STIME    FMRI
online         1:25:05  svc:/milestone/self-assembly-complete:default
online         1:25:09  svc:/milestone/multi-user-server:default
```

Displaying the Dependencies of a Service

To display service dependencies, run the `svcs -d` *FMRI* command.

```
# svcs -d svc:/network/ssh:default
STATE          STIME    FMRI
disabled       1:23:51  svc:/network/ipfilter:default
online         1:24:09  svc:/network/loopback:default
online         1:24:11  svc:/system/utmp:default
online         1:24:28  svc:/network/physical:default
online         1:24:36  svc:/system/filesystem/local:default
online         1:25:04  svc:/system/filesystem/autofs:default
```

Disabling a Service

1. Use the `svcs -D` *FMRI* command to check the dependents of the service that you want to disable.
2. Use the `svcadm disable` *FMRI* command to disable the service:

```
# svcadm disable svc:/network/ssh:default
```

3. Use the `svcs -l` *FMRI* command to verify that the service has been disabled:

```
# svcs -l svc:/network/ssh:default
fmri         svc:/network/ssh:default
name         SSH server
enabled      false
state        disabled
<output truncated>
```

Enabling a Service

1. Use the `svcs -l` *`FMRI`* `| grep online` command to determine whether service dependencies are satisfied.
2. Use the `svcadm enable` *`FMRI`* command to enable a service:

```
# svcadm enable svc:/network/ssh:default
```

3. Use the `svcs -x` *`FMRI`* command to verify that the service has been enabled:

```
# svcs -x svc:/network/ssh:default
svc:/network/ssh:default (SSH server)
 State: online since August 13, 2015 08:51:14 AM IST
   See: sshd(1M)
   See: /var/svc/log/network-ssh:default.log
Impact: None.
```

Refreshing and Restarting a Service

To refresh a service, run the `svcadm refresh` *FMRI* command.

```
# svcadm refresh svc:/network/ssh:default
```

To restart a service, run the `svcadm restart` *FMRI* command.

```
# svcadm restart svc:/network/ssh:default
```

Restoring a Service That Is in Maintenance State

1. Use the `svcs -x` *`FMRI`* command to determine why the service is in maintenance state.
2. Use the `svcs -l` *`FMRI`* command to determine if any process that is dependent to the service has not stopped.
3. Use the `svcs -o CTID` *`FMRI`* command to obtain the contract ID of the service that you want to restore.
4. Use `pkill -9 -c` *`CTID`* to kill any remaining processes.
5. Use `svcadm clear` *`FMRI`* to restore the service.

Setting Up Service State Transition Notifications

To set up the notifications:

1. Ensure that the `smtp-notify` package is installed
2. Enable the notification service
3. Configure the notifications

Monitored Transition States	
`to-uninitialized`	`to-disabled`
`from-uninitialized`	`from-disabled`
`to-maintenance`	`to-online`
`from-maintenance`	`from-online`
`to-offline`	`to-degraded`
`from-offline`	`from-degraded`

Installing the `smtp-notify` Package

Verify that the `system/fault-management/smtp-notify` package is already installed.

```
# pkg info system/fault-management/smtp-notify
```

If the package is not installed, run the following command to install the SMF notification feature:

```
# pkg install system/fault-management/smtp-notify
```

Enabling the `smtp-notify:default` Service

To enable the SMF notification service, run the following command:

```
# svcadm enable svc:/system/fm/smtp-notify:default
```

To confirm whether the service is up and running, run the following command:

```
# ps -ef | grep smtp-notify
noaccess  1060     1   0 11:45:9  ?       0:00 /usr/lib/fm/notify/smtp-notify
```

Configuring Service State Transition Notifications

- To configure service state transition notifications for all services, run the `svccfg -s svc:/system/svc/global:default setnotify -g` *`service_transition_state`* `mailto:root@localhost` command:

```
# svccfg -s svc:/system/svc/global:default setnotify -g from-online \
    mailto:root@localhost
```

- To configure notifications for a single service, run the `svccfg -s` *`FMRI`* `setnotify from-online mailto:root@localhost` command:

```
# svccfg -s svc:/network/http:apache22 setnotify from-online \
    mailto:root@localhost
```

Service State Transition Notification: Example

```
# mail
From noaccess@solaris.local Mon Aug 10 03:34:49 2015
Date: Mon, 10 Aug 2015 03:03:49 +0100 (CET)
From: No Access User
Message-Id: <201211090334.qA93YnCJ001559@s11-server1.example>
Subject: Fault Management Event: solaris:SMF-8000-YX
To: root@solaris.local
Content-Length: 776

SUNW-MSG-ID: SMF-8000-YX, TYPE: defect, VER: 1, SEVERITY: major
EVENT-TIME: Fri Aug 7 03:34:49 UTC 2015
PLATFORM: HVM domU, CSN: 26d2d7a7-1f3e-9df4-2bed-dc66344ea7e5, HOSTNAME: s11-
server1
SOURCE: software-diagnosis, REV: 0.1
EVENT-ID: 473a1ae7-5619-ea1e-dd03-8da51db4fcee
DESC: A service failed - a start, stop or refresh method failed.
AUTO-RESPONSE: The service has been placed into the maintenance state.
IMPACT: svc:/network/http:apache22 is unavailable.
REC-ACTION: Run 'svcs -xv svc:/network/http:apache22' to determine the
generic reason why the service failed, the location of any logfiles, and
a list of other services impacted. Please refer to the associated
reference document at http://support.oracle.com/msg/SMF-8000-YX for the
latest service procedures and policies regarding this diagnosis.

? <Press Enter to see the next message>
```

Service State Transition Notification: Example

```
<continued from previous slide>

From noaccess@localhost.example.com Mon Aug 10 03:34:21 2015
Date: Mon, 10 Aug 2015 03:34:21 GMT
From: No Access User <noaccess@s11-server1.example.com>
Message-Id: <201211090334.qA93YLum001539@s11-server1.example.com>
Subject: s11-server1: svc:/network/http:apache22 online->offline
To: root@s11-server1.example.com
Content-Length:776

HOSTNAME: s11-server1
TIMESTAMP: Mon Aug 10 12:04:23 2015
FMRI: svc:/network/http:apache22
FROM-STATE: online
TO-STATE: offline
DESCRIPTION: The indicated service has transitioned to the offline state
REASON: a restart was requested

? q
#
```

Managing Service State Transition Notifications

To view the configured notifications, run the following command:

```
# svccfg -s svc:/system/svc/global:default listnotify
    Event: from-online (source: svc:/system/svc/global:default)
        Notification Type: smtp
            Active: true
            to: root@localhost
```

To stop all notifications, run the following command:

```
# svccfg -s svc:/system/svc/global:default delnotify -g all
```

Quiz

Which of the following commands would you use to determine why a service is in maintenance state?

a. svcadm

b. svccfg

c. svcs

Quiz

Which of the following commands would you use to determine why a service is in maintenance state?

a. svcadm

b. svccfg

c. svcs

Summary

In this lesson, you should have learned how to:

- Describe the SMF feature and its components
- Administer SMF services

Practice 4: Overview

- 4-1: Administering Services
- 4-2: Administering SMF Notifications