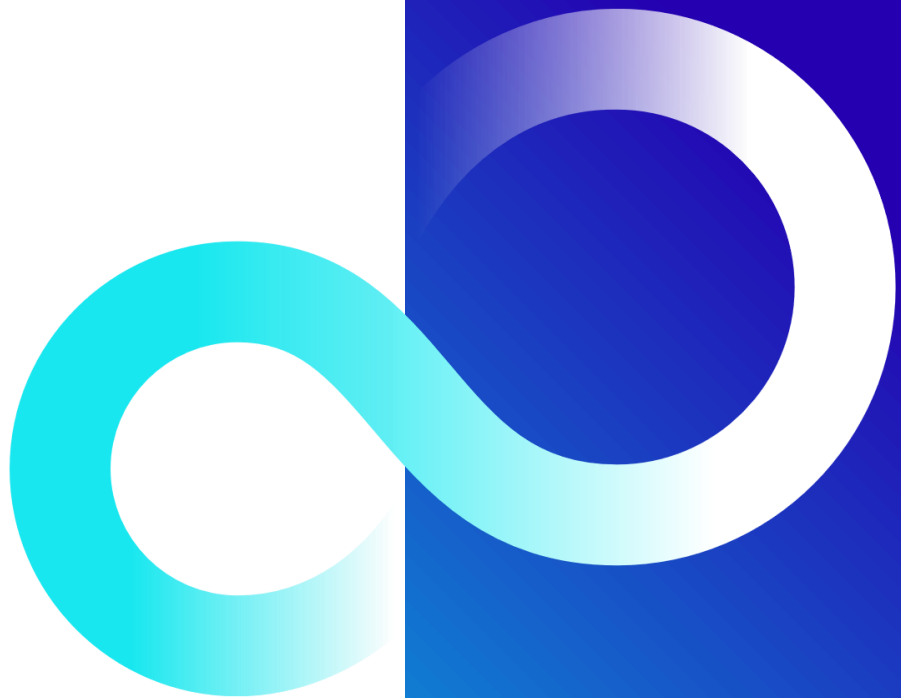


# Oracle Solaris 11 ゾーンを 使ってみよう

(構築・運用手順書)



2019年6月

第3.0版

富士通株式会社

## ■ 使用条件

- 著作権・商標権・その他の知的財産権について

コンテンツ(文書・画像・音声等)は、著作権・商標権・その他の知的財産権で保護されています。

本コンテンツは、個人的に使用する範囲でプリントアウトまたはダウンロードできます。ただし、これ以外の利用(ご自分のページへの再利用や他のサーバへのアップロード等)については、当社または権利者の許諾が必要となります。

- 保証の制限

本コンテンツについて、当社は、その正確性、商品性、ご利用目的への適合性等に関して保証するものではなく、そのご利用により生じた損害について、当社は法律上のいかなる責任も負いかねます。本コンテンツは、予告なく変更・廃止されることがあります。

- 輸出または提供

本製品を輸出又は提供する場合は、外国為替及び外国貿易法及び米国輸出管理関連法規等の規制をご確認の上、必要な手続きをおとり下さい。

## ■ 商標について

- UNIX は、米国およびその他の国におけるオープン・グループの登録商標です。
- SPARC Enterprise、SPARC64 およびすべての SPARC 商標は、米国 SPARC International, Inc.のライセンスを受けて使用している、同社の米国およびその他の国における商標または登録商標です。
- Oracle と Java は、Oracle Corporation およびその子会社、関連会社の米国およびその他の国における登録商標です。
- その他各種製品名は、各社の製品名称、商標または登録商標です。

## はじめに

### 本書の内容

- Oracle Solaris 11 ゾーンの構築・運用手順を紹介しています。別冊の『Oracle Solaris 11 ゾーンを使ってみよう(構築・運用ガイド)』を参照しながら、本手順書を活用してください。

### 留意事項

- 本資料は、Oracle Solaris 11.4 を使用した手順を紹介しています。
- 本手順書に記載しているログの値は、環境によって異なることがあります。

### 本書での表記

- 以下の用語は略称を用いて表記する場合があります。

| 略称          | 正式名称   |
|-------------|--|
| Solaris     | Oracle Solaris                                   |
| Solaris ゾーン | Oracle Solaris ゾーン<br>(旧名称: Oracle Solaris コンテナ) |
| Oracle VM   | Oracle VM Server for SPARC                       |
| OBP         | Open Boot Prom                                   |
| IPS         | Image Packaging System                           |

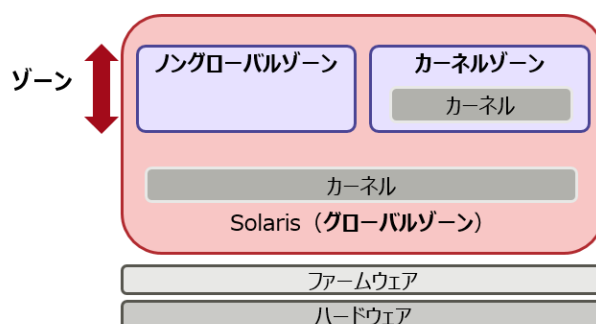
- 本書のコマンド例では、実行する環境によって以下のプロンプトで表記しています。

| 実行する環境     | プロンプト                  |
|------------|------------------------|
| グローバルゾーン   | global01#<br>global02# |
| ノングローバルゾーン | zone01#<br>zone02#     |

## 本書で使用する Oracle Solaris ゾーン関連の用語

- Solaris ゾーンとは、Solaris の仮想環境を提供するためのサーバ仮想化機能の 1 つです。
- Solaris ゾーンを用いて作成できる Solaris の仮想環境をゾーンと呼びます。
- ゾーンには、ノングローバルゾーンとカーネルゾーンの 2 種類があります。

| ゾーンの種類     | 説明   |
|------------|--|
| ノングローバルゾーン | グローバルゾーン(*1)とカーネルを共有する仮想環境です。<br>※ドキュメントによっては、「non-global zone」と記載しています。 |
| カーネルゾーン    | ゾーンごとに独立したカーネルを持つ仮想環境です。<br>※Solaris 11.2 からサポートされています。                  |



\*1: グローバルゾーンとは、物理サーバ上で動作する OS 環境です。グローバルゾーンから、ゾーンの設定や制御をします。

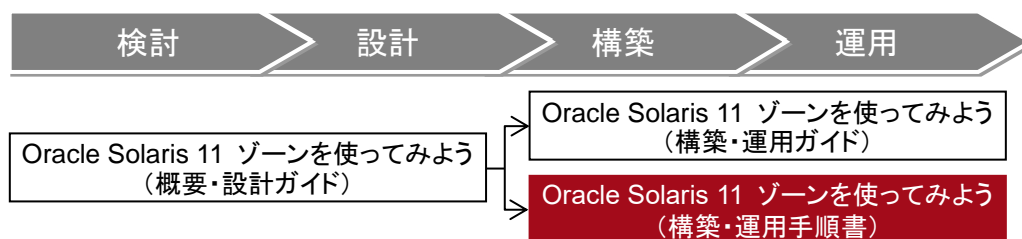
## Point

本書では、「ノングローバルゾーン」を「ゾーン」と表記している箇所があります。

## ドキュメントの位置づけ

- Oracle Solaris 11 ゾーンを使ってみよう

<https://www.fujitsu.com/jp/sparc-technical/document/solaris/#solaris-zone>



※ 本書は、『Oracle Solaris 11 ゾーンを使ってみよう(構築・運用ガイド)』とあわせてお読みください。

## 目次

|                                     |    |
|-------------------------------------|----|
| 1. リソースプールの作成 .....                 | 1  |
| 1.1. スケジューラの設定 .....                | 1  |
| 1.2. サービスの起動 .....                  | 3  |
| 1.2.1. パッケージの確認 .....               | 3  |
| 1.2.2. リソースプールサービスの起動 .....         | 3  |
| 1.3. リソースプールの設定 .....               | 5  |
| 1.4. リソースプールの作成 .....               | 9  |
| 2. ゾーン(仮想 Solaris 環境)の作成 .....      | 13 |
| 2.1. 事前準備 .....                     | 13 |
| 2.1.1. ゾーンのパッケージ確認 .....            | 13 |
| 2.1.2. ゾーンの状態確認 .....               | 14 |
| 2.1.3. ゾーン用の新規ストレージプールの作成 .....     | 14 |
| 2.1.4. マウントポイントの確認 .....            | 14 |
| 2.2. ゾーンの設定 .....                   | 14 |
| 2.3. ゾーンのインストール .....               | 22 |
| 2.4. ゾーンの起動 .....                   | 25 |
| 2.5. 仮想 OS の初期設定 .....              | 26 |
| 3. ゾーン(仮想 Solaris 環境)の基本操作 .....    | 33 |
| 3.1. ゾーンのスレータスの確認 .....             | 33 |
| 3.2. ゾーンの起動／停止 .....                | 34 |
| 3.2.1. ゾーンの起動・再起動・停止操作(通常運用時) ..... | 34 |
| 3.2.2. ゾーンの起動・再起動・停止操作(緊急時) .....   | 36 |
| 3.3. ゾーンへのログイン .....                | 36 |
| 3.4. ゾーンへのコマンド実行 .....              | 37 |
| 3.5. ゾーン情報の確認 .....                 | 38 |
| 3.5.1. グローバルゾーンでのゾーン情報の確認 .....     | 38 |
| 3.5.2. ノングローバルゾーンでのゾーン情報の確認 .....   | 39 |
| 3.6. ゾーンの構成情報の変更 .....              | 40 |
| 3.6.1. ゾーンの停止 .....                 | 40 |

|        |                                      |    |
|--------|--------------------------------------|----|
| 3.6.2. | ゾーン構成情報の変更 .....                     | 41 |
| 3.6.3. | ゾーンの起動 .....                         | 42 |
| 3.6.4. | 設定したプロパティの削除 .....                   | 43 |
| 3.7.   | ゾーンの統計情報の確認 .....                    | 46 |
| 4.     | ゾーン(仮想 Solaris 環境)の複製 .....          | 47 |
| 4.1.   | 構成情報の作成 .....                        | 47 |
| 4.2.   | クローン(複製)の作成 .....                    | 49 |
| 4.3.   | クローン(複製)の初期設定 .....                  | 49 |
| 5.     | ゾーン(仮想 Solaris 環境)の移動 .....          | 51 |
| 5.1.   | 統合アーカイブを使用したゾーンの移動 .....             | 51 |
| 5.1.1. | リカバリーアーカイブの作成と転送 .....               | 52 |
| 5.1.2. | リカバリーアーカイブの展開とゾーンのインストール .....       | 53 |
| 5.2.   | ZFS アーカイブを使用したゾーンの移動 .....           | 56 |
| 5.2.1. | ゾーンの切り離しと構成情報の抽出 .....               | 56 |
| 5.2.2. | アーカイブの作成と転送 .....                    | 57 |
| 5.2.3. | アーカイブの展開とゾーンの組み込み .....              | 59 |
| 6.     | ゾーン(仮想 Solaris 環境)の CPU リソース操作 ..... | 62 |
| 6.1.   | CPU シェア数の設定 .....                    | 62 |
| 6.1.1. | リソース操作の事前準備と確認 .....                 | 62 |
| 6.1.2. | CPU シェア数の設定 .....                    | 63 |
| 6.2.   | CPU キャップの設定 .....                    | 64 |
| 6.3.   | CPU リソースの動的移動 .....                  | 66 |
| 7.     | ゾーン(仮想 Solaris 環境)の仮想ネットワークの応用 ..... | 68 |
| 7.1.   | 事前準備 .....                           | 68 |
| 7.2.   | etherstub の設定 .....                  | 68 |
| 7.2.1. | etherstub の作成 .....                  | 68 |
| 7.2.2. | ゾーンに etherstub を割り当て .....           | 69 |
| 7.2.3. | ゾーン相互の疎通確認 .....                     | 71 |
| 7.3.   | IPMP の設定 .....                       | 72 |

|                       |    |
|-----------------------|----|
| 7.3.1. ゾーン環境の準備 ..... | 72 |
| 7.3.2. IPMP の作成 ..... | 74 |
| 7.4. 帯域幅制限の設定 .....   | 76 |
| 7.4.1. ゾーン環境の準備 ..... | 76 |
| 7.4.2. フローの作成 .....   | 76 |
| 7.5. VLAN の設定 .....   | 79 |
| 7.5.1. ゾーン環境の準備 ..... | 79 |
| 7.5.2. VLAN の作成 ..... | 80 |
| 改版履歴 .....            | 84 |

## ゾーン構築にあたって

ゾーンの構築のみ実行する場合は、「[1. リソースプールの作成](#)」および「[2. ゾーン\(仮想 Solaris 環境\)の作成](#)」を実施してください。なお、CPU リソースの配分を設定しない場合は、「[2. ゾーン\(仮想 Solaris 環境\)の作成](#)」のみ実施してください。

※ 「[2. ゾーン\(仮想 Solaris 環境\)の作成](#)」のみ実施する場合は、リソースプールがグローバルゾーンと同じ「pool\_default」になります。

「[3. ゾーン\(仮想 Solaris 環境\)の基本操作](#)」以降は、ゾーンの操作や機能の紹介です。

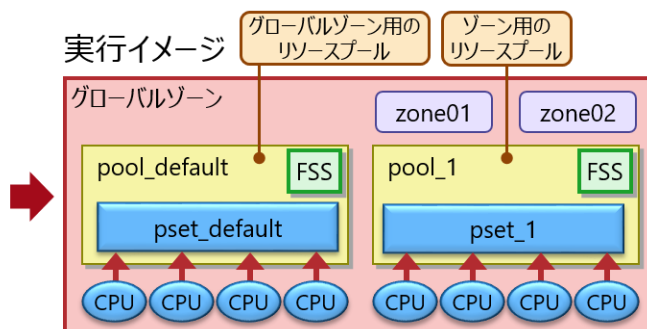


## 1. リソースプールの作成

リソースプールの設定項目の設定例、および実行イメージを以下に示します。

リソースプール設定項目

|            | グローバルゾーン用    | ゾーン用   |
|------------|--------------|--------|
| リソースプール    | pool_default | pool_1 |
| スケジューラ     | FSS          | FSS    |
| プロセスセット名   | pset_default | pset_1 |
| 最大CPUスレッド数 | 65535        | 4      |
| 最小CPUスレッド数 | 4            | 4      |



### 1.1. スケジューラの設定

#### 1) プロセスのスケジューラを確認します。

デフォルトで動作している init プロセスのスケジューラが、「TS(タイムシェアリング)」であることを確認します。

```
global01# ps -efc | grep /sbin/init
root      1      0    TS   59   Dec 05 ?           0:01 /usr/sbin/init
root 22348 22339    TS   49 15:22:46 console 0:00 grep /sbin/init
```

#### 2) デフォルトのスケジューラを確認します。

```
global01# dispadmin -d
dispadmin: Default scheduling class is not set
```

▶ OS 起動時のスケジューラ設定を確認するコマンドです。スケジューラが未設定の場合は、確認できません。

#### 3) デフォルトのスケジューラを変更します。

本書では、OS のデフォルトのスケジューラを「FSS(Fair Share Scheduler)」に変更します。

```
global01# dispadmin -d FSS
```

#### 4) デフォルトのスケジューラが変更されたことを確認します。

「FSS」に変更されたことを確認します。

```
global01# dispadmin -d
FSS      (Fair Share)
```

▶ サーバの再起動または、priocntl コマンドによる変更をするまでは、プロセスのスケジューラは変更されません。

#### 5) サーバを再起動します。

```
global01# shutdown -y -g0 -i6
```

#### 6) 一般ユーザーでログインし、root 権限へ切り替えます。



## 7) プロセスのスケジューラが変更されたことを確認します。

「FSS」に変更されたことを確認します。

```
global01# ps -efc | grep /sbin/init
root      1      0  FSS  59 15:27:58 ?          0:00 /usr/sbin/init
root    1263    1252  FSS  59 15:29:21 console    0:00 grep /sbin/init
```

### 《参考》再起動しない場合の手順

以下の手順を実施することにより、再起動なしでプロセスのスケジューラを変更できます。

(1) init プロセスのスケジューラを「FSS」に変更します。

```
global01# priocntl -s -c FSS -i pid 1
```

(2) TS クラスで動作しているプロセスのスケジューラを「FSS」に変更します。

```
global01# priocntl -s -c FSS -i class TS
```

### 《注意》

サーバを再起動すると、プロセスのスケジューラは変更前に戻ります（本書の場合は「TS」に戻ります）。

再起動後もプロセスのスケジューラを「FSS」にする場合は、デフォルトのスケジューラを「FSS」に変更します。

```
global01# dispadmin -d FSS
```

### 《参考》スケジューラの初期化

デフォルトのスケジューラを「TS」に戻します。

```
global01# dispadmin -d TS
global01# init 6
```

## 1.2. サービスの起動

### 1.2.1. パッケージの確認

#### 1) リソースプールのパッケージを確認します。

パッケージがインストール済みの場合、「Installed」と表示されます。

```
global01# pkg info system/resource-mgmt/resource-pools
Name: system/resource-mgmt/resource-pools
Summary: Resource pools
Description: Resource pools support the separation of workloads so that
workload consumption of certain resources does not overlap
and so predictable performance on systems with mixed
workloads can be better achieved. Resource pools provide a
mechanism for persistent processor set (pset) configuration
and, optionally, scheduling class assignment.
Category: System/Core
State: Installed
Publisher: solaris
Version: 11.4
Branch: 11.4.6.0.1.4.0
Packaging Date: February 1, 2019 at 9:33:58 PM
Last Install Time: March 26, 2019 at 1:30:23 AM
Size: 963.11 kB
FMRI: pkg://solaris/system/resource-mgmt/resource-pools@11.4-
11.4.6.0.1.4.0:20190201T213358Z
```

### 1.2.2. リソースプールサービスの起動

#### 1) リソースプールサービスの状態を確認します。

「svc:/system/pools:default」サービスが「disabled」（無効）と表示されることを確認します。

```
global01# svcs -a | grep pools
disabled      9:12:13 svc:/system/pools:default
disabled      9:12:16 svc:/system/pools/dynamic:default
```

#### 2) リソースプールサービスを起動します。

```
global01# svcadm enable svc:/system/pools:default
```

#### 3) リソースプールサービスの状態を確認します。

「svc:/system/pools:default」サービスが「online」（有効）と表示されることを確認します。

```
global01# svcs -a | grep pools
disabled      9:12:16 svc:/system/pools/dynamic:default
online        16:08:14 svc:/system/pools:default
```

#### 4) リソースプールの状態を確認します。

構成されているリソースプールと pset、および搭載されている CPU スレッドを確認します。

```
global01# pooladm

system default
  string system.comment
  int    system.version 1
  boolean system.project-fallback-to-default true
  boolean system.bind-default true
  boolean system.zone-fallback-to-default false
  string system.poolid.objectives wt-load

pool pool_default ← pool は、pool_default のみ存在することを確認
  int    pool.sys_id 0
  boolean pool.active true
  boolean pool.default true
  int    pool.importance 1
  string pool.comment
  boolean pool.temporary false
  boolean pool.psrset false
  pset   pset_default

pset pset_default ← pset は、pset_default のみ存在することを確認
  int    pset.sys_id -1
  boolean pset.default true
  uint   pset.min 1
  uint   pset.max 65536
  string pset.policy minmax
  string pset.restype cpu
  string pset.reslist
  string pset.units population
  uint   pset.load 6
  uint   pset.size 48
  string pset.comment
  boolean pset.temporary false
  boolean pset.psrset false

cpu ← 搭載 CPU スレッドの数だけ CPU が存在することを確認
  int    cpu.sys_id 37
  string cpu.comment
  string cpu.status on-line

~（省略）~
```

- ▶ 初期状態では、リソースプールはグローバルゾーン専用のリソースプール(pool\_default)が存在します。
- ▶ SPARC M12-1 の物理サーバの場合は、48 スレッド分表示されます(cpu.sys\_id 0 - 47)。

## 5) poolstat コマンドでリソースプールの状態を確認します。

```
global01# poolstat -r all
```

| id | pool         | type | rid | rset         | min | max | size | used | load |
|----|--------------|------|-----|--------------|-----|-----|------|------|------|
| 0  | pool_default | pset | -1  | pset_default | 1   | 66K | 48   | 0.00 | 0.01 |

- ▶ pool: リソースプール名
- rset: リソースプールに関連付けされた pset 名
- min: 割り当てられる CPU スレッドの最大数
- max: 割り当てられる CPU スレッドの最小数
- size: 割り当てされた CPU スレッド数

## 1.3. リソースプールの設定

### 1) リソースプール構成ファイルを作成します。

作成されたファイルにリソースプール構成を保存します。本書は、ファイル名を指定しないことでデフォルトのリソースプール構成ファイル「/etc/pooladm.conf」を作成します。

すでに「/etc/pooladm.conf」が存在する場合、本手順は実施不要です。

【書式】 pooladm -s <ファイル名>

```
global01# pooladm -s
```

## 2) リソースプール構成ファイルの内容を確認します。

pool\_defaultとpset\_defaultのみの構成であることを確認します。

```
global01# poolcfg -c info

system default
  string  system.comment
  int     system.version 1
  boolean system.project-fallback-to-default true
  boolean system.bind-default true
  boolean system.zone-fallback-to-default false
  string  system.poolid.objectives wt-load

pool pool_default
  int     pool.sys_id 0
  boolean pool.active true
  boolean pool.default true
  int     pool.importance 1
  string  pool.comment
  boolean pool.temporary false
  boolean pool.psrset false
  pset    pset_default

pset pset_default
  int     pset.sys_id -1
  boolean pset.default true
  uint    pset.min 1
  uint    pset.max 65536
  string  pset.policy minmax
  string  pset.restype cpu
  string  pset.reslist
  string  pset.units population
  uint    pset.load 7
  uint    pset.size 48
  string  pset.comment
  boolean pset.temporary false
  boolean pset.psrset false

cpu
  int     cpu.sys_id 37
  string  cpu.comment
  string  cpu.status on-line

~（省略）~
```

### 3) リソースプール構成ファイルを編集します。

- デフォルトのプロセッサセットの最小 CPU スレッド数を変更

デフォルトのプロセッサセット(pset\_default)の最小 CPU スレッド数を 4 に変更します。

```
global01# poolcfg -c 'modify pset pset_default (uint pset.min=4)'
```

- 新規プロセッサセット(pset)の作成

新規プロセッサセット「pset\_1」を作成し、最大 CPU スレッド数と最小 CPU スレッド数を 4 に設定します。

```
global01# poolcfg -c 'create pset pset_1 (uint pset.min=4; uint pset.max=4)'
```

- 新規リソースプール(pool)の作成

新規リソースプール「pool\_1」を作成します。

```
global01# poolcfg -c 'create pool pool_1'
```

- プロセッサセットとリソースプールの関連付け

プロセッサセット「pset\_1」とリソースプール「pool\_1」を関連付けます。

```
global01# poolcfg -c 'associate pool pool_1 (pset pset_1)'
```

- リソースプールのスケジューラ設定

リソースプール「pool\_1」のスケジューラを「FSS」に設定します。

```
global01# poolcfg -c 'modify pool pool_1 (string pool.scheduler="FSS")'
```

**《参考》一括定義方法**

リソースプールの設定は編集済みファイルを読み込ませることで、一括で定義することもできます。

(1) 任意のファイルに定義情報を記述します。

```
global01# vi /var/tmp/poolcfg.txt
```

(記載例)

```
create system default
modify pset pset_default (uint pset.min=4)
create pset pset_1 (uint pset.min=4; uint pset.max=4)
create pool pool_1
associate pool pool_1 (pset pset_1)
modify pool pool_1 (string pool.scheduler="FSS")
```

(2) ファイルを読み込みます。

```
global01# poolcfg -f /var/tmp/poolcfg.txt
```

(3) 構成を実行します。

```
global01# pooladm -c
```

**《参考》リソースプールの設定を変更するには**

「[《参考》リソースプールの設定の変更・削除](#)」を参照してください。

## 1.4. リソースプールの作成

### 1) リソースプール構成ファイルの内容を確認します。

作成した「pool\_1」と「pset\_1」が追加されていることを確認します。

```
global01# poolcfg -c info

system default
  string  system.comment
  int     system.version 1
  boolean system.project-fallback-to-default true
  boolean system.bind-default true
  boolean system.zone-fallback-to-default false
  string  system.poolid.objectives wt-load

pool pool_default
  int     pool.sys_id 0
  boolean pool.active true
  boolean pool.default true
  int     pool.importance 1
  string  pool.comment
  boolean pool.temporary false
  boolean pool.psrset false
  pset    pset_default

pool pool_1
  boolean pool.active true
  boolean pool.default false
  string  pool.scheduler FSS
  int     pool.importance 1
  string  pool.comment
  boolean pool.temporary false
  boolean pool.psrset false
  pset    pset_1

pset pset_default
  int     pset.sys_id -1
  boolean pset.default true
  uint    pset.min 4
  uint    pset.max 65536
  string  pset.policy minmax
  string  pset.restype cpu
  string  pset.reslist
  string  pset.units population
  uint    pset.load 7
  uint    pset.size 48
  string  pset.comment
  boolean pset.temporary false
```



```

boolean pset.psrset false

cpu
    int    cpu.sys_id 37
    string  cpu.comment
    string  cpu.status on-line

~（省略）~

cpu
    int    cpu.sys_id 10
    string  cpu.comment
    string  cpu.status on-line

pset pset_1
    int    pset.sys_id -2
    boolean pset.default false
    uint    pset.min 4
    uint    pset.max 4
    string  pset.policy minmax
    string  pset.restype cpu
    string  pset.reslist
    string  pset.units population
    uint    pset.load 0
    uint    pset.size 0
    string  pset.comment
    boolean pset.temporary false
    boolean pset.psrset false

```

## 2) 構成を実行します。

リソースプール構成ファイルに設定した内容を反映します。

```
global01# pooladm -c
```

▶ 実行時にエラーが発生する場合は、構成ファイルを見直してください。

## 3) 実行中のリソースプール構成を確認します。

作成した「pool\_1」と「pset\_1」が出力されていること、および「pset\_default」の最小 CPU スレッド数が変更されていることを確認します。

```
global01# poolstat -r all
```

| id | pool         | type | rid | rset         | min | max | size | used | load |
|----|--------------|------|-----|--------------|-----|-----|------|------|------|
| 1  | pool_1       | pset | 1   | pset_1       | 4   | 4   | 4    | 0.00 | 0.00 |
| 0  | pool_default | pset | -1  | pset_default | 4   | 66K | 44   | 0.00 | 0.04 |

## 《参考》 リソースプールの初期化と停止

### 《注意》

リソースプールは、Oracle Solaris ゾーン起動中に初期化しないでください。  
ゾーンに設定されたリソースプールが、デフォルトの「pool\_default」に変更されます。

- 現在動作しているリソースプール状態を破棄（初期状態の設定に戻る）

リソースプール構成を初期化します。

```
global01# pooladm -x
```

▶ リソースプールは、「pool\_default」および「pset\_default」のみの状態になります。

- pset\_default の設定の初期化

本書では、pset\_default の最小 CPU スレッド数を変更しているため、元の設定に戻します。

なお、Oracle Solaris 11.4 では pooladm -x で初期化されるため、本操作は不要です。

```
global01# poolcfg -c 'modify pset pset_default (uint pset.min=1)'
```

- リソースプール構成の保存

初期化された構成をリソースプール構成ファイルに保存します。

```
global01# pooladm -s
```

- 現在動作しているリソースプールの停止（リソースプールサービスの停止）

リソースプールサービスを停止します。

```
global01# svcadm disable svc:/system/pools:default
global01# svcadm disable svc:/system/pools/dynamic:default
```

リソースプールサービスが停止していることを確認します。

```
global01# svcs -a | grep pools
disabled    10:53:34 svc:/system/pools:default
disabled    10:53:36 svc:/system/pools/dynamic:default
```

**《参考》 リソースプールの設定の変更・削除**

リソースプールの設定を変更・削除したあとは、pooladm -c コマンドで適用してください。

- 設定した名前の変更

例: プール「pool\_1」を「pool\_2」に変更

```
global01# poolcfg -c 'rename pool pool_1 to pool_2'
```

例: プロセッサセット「pset\_1」を「pset\_2」に変更

```
global01# poolcfg -c 'rename pset pset_1 to pset_2'
```

- 設定したプロパティの変更

例: プール「pool\_1」のコメントを変更

```
global01# poolcfg -c 'modify pool pool_1 (string pool.comment="TEST_POOL")'
```

例: プロセッサセット「pset\_1」の最大 CPU スレッド数を「8」に変更

```
global01# poolcfg -c 'modify pset pset_1 (uint pset.max=8)'
```

- 設定したプロパティの削除

例: プール「pool\_1」のスケジューラ設定を削除

```
global01# poolcfg -c 'modify pool pool_1 (~ string pool.scheduler)'
```

- 設定の削除

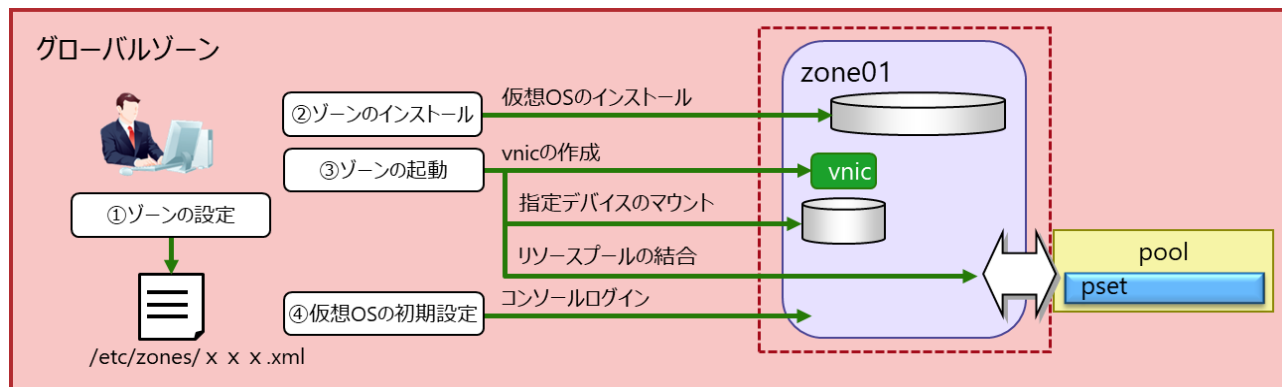
例: プール「pool\_1」を削除

```
global01# poolcfg -c 'destroy pool pool_1'
```

例: プロセッサセット「pset\_1」を削除

```
global01# poolcfg -c 'destroy pset pset_1'
```

## 2. ゾーン(仮想 Solaris 環境)の作成



### 2.1. 事前準備

#### 2.1.1. ゾーンのパッケージ確認

必要なパッケージがインストールされているか確認します。

パッケージがインストール済みの場合、「Installed」と表示されます。

```
global01# pkg info system/zones
Name: system/zones
Summary: Solaris Zones configuration and administration
Description: The Solaris Zones partitioning technology is used to
virtualize operating system services and provide an isolated,
secure, lightweight, and high performance environment for
running applications. A zone is a virtualized operating
system environment created within a single instance of the
operating system.
Category: System/Core
State: Installed
Publisher: solaris
Version: 11.4
Branch: 11.4.6.0.1.4.0
Packaging Date: February 1, 2019 at 9:34:42 PM
Last Install Time: March 26, 2019 at 1:30:23 AM
Size: 10.21 MB
FMRI: pkg://solaris/system/zones@11.4-11.4.6.0.1.4.0:20190201T213442Z
```

### 2.1.2. ゾーンの状態確認

ゾーンの状態一覧を表示します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS  | PATH | BRAND   | IP     |
|----|--------|---------|------|---------|--------|
| 0  | global | running | /    | solaris | shared |

▶ 初期状態では、グローバルゾーン(global)のみが表示されます。

### 2.1.3. ゾーン用の新規ストレージプールの作成

ここでは例として、ゾーン用ストレージプール「zones」をディスク「c1t1d4」に作成します。

【書式】zpool create <プール名> <ディスク>

```
global01# zpool create zones c1t1d4
global01# zpool list
```

| NAME  | SIZE  | ALLOC | FREE  | CAP | DEDUP | HEALTH | ALTROOT |
|-------|-------|-------|-------|-----|-------|--------|---------|
| rpool | 556G  | 72.7G | 483G  | 13% | 1.00x | ONLINE | -       |
| zones | 49.8G | 118K  | 49.7G | 0%  | 1.00x | ONLINE | -       |

▶ 本書ではゾーン用の領域として/zonesを使用します。

### 2.1.4. マウントポイントの確認

```
global01# zfs list
```

| NAME                     | USED  | AVAIL | REFER | MOUNTPOINT          |
|--------------------------|-------|-------|-------|---------------------|
| rpool                    | 72.7G | 475G  | 73.5K | /rpool              |
| ~ (省略) ~                 |       |       |       |                     |
| rpool/export/home/user01 | 33.5K | 475G  | 33.5K | /export/home/user01 |
| rpool/swap               | 4.00G | 475G  | 4.00G | -                   |
| zones                    | 100K  | 49.0G | 31K   | /zones              |

## 2.2. ゾーンの設定

### 1) ゾーン構成を開始します。

```
global01# zonecfg -z zone01
Use 'create' to begin configuring a new zone.
zonecfg:zone01>
```

▶ 実行後プロンプトが変わり、構成コマンドの入力が可能となります。

### 2) ゾーンを作成します。

```
zonecfg:zone01> create
create: Using system default template 'SYSdefault'
zonecfg:zone01>
```

▶ 初期設定値を読み込みます。ゾーン作成時に必ず最初に実行する必要があります。

### 3) ゾーン設定を表示します。

ゾーンの設定内容を確認します。

```
zonecfg:zone01> info -a
zonename: zone01
zonepath.template: /system/zones/{zonename}
zonepath: /system/zones/zone01
brand: solaris
autoboot: false
autoshutdown: shutdown
bootargs:
file-mac-profile:
pool:
limitpriv:
scheduling-class:
ip-type: exclusive
hostid:
tenant:
fs-allowed:
anet 0:
    linkname: net0
    lower-link: auto
    allowed-address:
    configure-allowed-address: true
    defrouter:
    allowed-dhcp-cids:
    link-protection: mac-nospoof
    mac-address: auto
    auto-mac-address:
    mac-prefix:
    mac-slot:
    vlan-id:
    priority:
    rxrings:
    txrings:
    mtu:
    maxbw:
    bwshare:
    rxfanout:
    vsi-typeid:
    vsi-vers:
    vsi-mgrid:
    etsbw-lcl:
    cos:
    pkey:
    linkmode:
    evs:
    vport:
```

```
ring-group: auto
autopush:
```

▶ `-a` オプションを使用しない場合は、デフォルト値以外の値を持つプロパティのみ表示します。

#### 4) ゾーンパスを設定します。

ゾーン環境のルートディレクトリを、グローバルゾーン上の絶対パスで指定します。

```
zonecfg:zone01> set zonepath=/zones/zone01
```

▶ `zonepath` で指定したフォルダは自動的に作成されます。

#### 5) ゾーンの自動起動を設定します。

```
zonecfg:zone01> set autoboot=false
```

#### 6) リソースプールの関連付けを設定します。

ゾーンで使用するリソースプールを指定します。

##### 《注意》

本設定は、「[1. リソースプールの作成](#)」を実施してリソースプールを作成した状態で実施してください。リソースプールが未作成の状態で設定を行うと、ゾーン起動時にエラーが発生します。

「[1. リソースプールの作成](#)」を実施しない（リソースプールを作成しない）場合は、グローバルゾーンと同じ「`pool_default`」が設定されます。

```
zonecfg:zone01> set pool=pool_1
```

##### 《参考》 設定を解除するには

以下のコマンドを実行してください。

```
zonecfg:zone01> clear pool
```

#### 7) スケジューラを設定します。

ゾーンで使用するリソースプールのスケジューラを指定します。本書では、「FSS」を設定します。

```
zonecfg:zone01> set scheduling-class=FSS
```

#### 8) ネットワークインターフェースの排他設定をします。

本書では、`anet` リソースを使用するため、`ip-type` を `exclusive` に設定します。

なお、Oracle Solaris 11 では、デフォルトで「`exclusive`」に設定されているため、下記は実施しなくても問題ありません。

```
zonecfg:zone01> set ip-type=exclusive
```

## 9) VNICを確認します。

Oracle Solaris 11 では、デフォルトで anet パラメーターが設定されています。anet は、ゾーン起動時に自動的に仮想 NIC を作成し、ゾーンに割り当てる機能です。

anet は、ネットワークインターフェースが占有設定 (ip-type=exclusive) の場合のみ使用できます。共有設定 (ip-type=shared) では使用できません。

```
zonecfg:zone01> info -a anet
anet 0:
    linkname: net0
    lower-link: auto
    allowed-address:
    configure-allowed-address: true
    defrouter:
    allowed-dhcp-cids:
    link-protection: mac-nospoof
    mac-address: auto
    auto-mac-address:
    mac-prefix:
    mac-slot:
    vlan-id:
    priority:
    rxrings:
    txrings:
    mtu:
    maxbw:
    bwshare:
    rxfanout:
    vsi-typeid:
    vsi-vers:
    vsi-mgrid:
    etsbw-lcl:
    cos:
    pkey:
    linkmode:
    evs:
    vport:
    ring-group: auto
    autopush:
```

## 10) anet リソースの linkname プロパティを変更します。

linkname には、ゾーン環境上で表示するインターフェース名を設定します。

```
zonecfg:zone01> select anet linkname=net0
zonecfg:zone01:anet> set linkname=vnic0
```

- ▶ 変更しない場合は、デフォルトの「net0」となります。



## 11) anet リソースの lower-link プロパティを変更します。

lower-link には、anet に紐づけるグローバルゾーン上のインターフェース名を設定します。ここで設定したインターフェース上に仮想 NIC が作成されます。

デフォルトは「auto（自動設定）」ですが、ここでは「net0」を設定します。

```
zonecfg:zone01:anet> set lower-link=net0
zonecfg:zone01:anet> end
```

### 《参考》 グローバルゾーンとディレクトリを共有するには

共有ファイルシステムの設定を行います。

```
zonecfg:zone01> add fs
zonecfg:zone01:fs> set dir=/cdrom
zonecfg:zone01:fs> set special=/cdrom
zonecfg:zone01:fs> set type=lofs
zonecfg:zone01:fs> add options [ro,nodevices]
zonecfg:zone01:fs> end
zonecfg:zone01>
```

- ▶ グローバルゾーンの/cdrom ディレクトリを共有します。
- ▶ share ディレクトリとなる special プロパティは、グローバルゾーンに存在するディレクトリを指定します。  
マウントポイントとなる dir プロパティは、ノングローバルゾーンにマウントさせるディレクトリを指定します。
- ▶ ゾーン起動後に設定する場合は、zoneadm apply コマンドで起動中のノングローバルゾーンに適用可能です。  
※ 詳細は、『《参考》 ゾーンの設定の変更・削除』を参照してください。  
※ ノングローバルゾーンの再起動でも適用可能です。

### 《参考》 MAC アドレスを直接指定するには

利用可能な MAC アドレスを確認して指定します。

```
global01# dladm show-phys -m
LINK          SLOT      ADDRESS          INUSE CLIENT
net0          primary  0:14:4f:f9:65:4e yes  net0
              1        0:14:4f:fb:f:ae  no   --
              2        0:14:4f:fb:13:6  no   --

global01# zonecfg -z zone01
zonecfg:zone01> create
zonecfg:zone01> select anet mac-address=auto
zonecfg:zone01:anet> set mac-address=0:14:4f:fb:f:ae
zonecfg:zone01:anet> end
```

※ Solaris 11 では、Oracle VM のゲストメイン環境上に anet リソースでゾーンを構築する場合、mac-address プロパティには「auto」を設定するか、MAC アドレスを直接指定する必要があります。詳細は、以下を参照してください。

『ネットワーク設計・構築ガイド～on Oracle VM Server for SPARC～』

<https://www.fujitsu.com/jp/sparc-technical/document/solaris/#solaris-zone>

## 12) ゾーンの設定を確認します。

```
zonecfg:zone01> info -a
zonename: zone01
zonepath: /zones/zone01
brand: solaris
autoboot: false
autosshutdown: shutdown
bootargs:
file-mac-profile:
pool: pool_1
limitpriv:
scheduling-class: FSS
ip-type: exclusive
hostid:
tenant:
fs-allowed:
anet 0:
    linkname: vnic0
    lower-link: net0
    allowed-address:
    configure-allowed-address: true
    defrouter:
    allowed-dhcp-cids:
    link-protection: mac-nospoof
    mac-address: auto
    auto-mac-address:
    mac-prefix:
    mac-slot:
    vlan-id:
    priority:
    rxrings:
    txrings:
    mtu:
    maxbw:
    bwshare:
    rxfanout:
    vsi-typeid:
    vsi-vers:
    vsi-mgrid:
    etsbw-lcl:
    cos:
    pkey:
    linkmode:
    evs:
    vport:
    ring-group: auto
    autopush:
```

**《参考》 デバイスの共有設定 (add device プロパティの設定例)**

- デバイスの共有

例として、テープドライブの設定方法を、以下に示します。

```
zonecfg:zone01> add device
zonecfg:zone01:device> set match=/dev/rmt/0*
zonecfg:zone01:device> end
zonecfg:zone01>
```

▶ テープデバイス名を指定し、デバイス名の後ろに「\*」を付けます。

- format コマンドによるディスクのラベル付けを可能に設定

例として、ディスク「c2t0d1」のラベル付けを可能にする場合の設定方法を、以下に示します。

```
zonecfg:zone01> add device
zonecfg:zone01:device> set match=/dev/*dsk/c2t0d1*
zonecfg:zone01:device> set allow-partition=true
zonecfg:zone01:device> end
zonecfg:zone01>
```

▶ ディスクまたは LUN 全体を指定し、allow-partition プロパティを有効にします。

ゾーン上でデータベースソフトウェアを使用する場合などは、以下の設定が必要になる場合があります。必要に応じて実施してください。

- raw デバイスを使用可能に設定 (raw デバイスを扱う場合)

```
zonecfg:zone01> add device
zonecfg:zone01:device> set match=/dev/*dsk/cNtXdY*
zonecfg:zone01:device> end
zonecfg:zone01>
```

- UFS をマウント可能に設定 (UFS ファイルシステムを扱う場合)

```
zonecfg:zone01> set fs-allowed=ufs
```

▶ fs-allowed プロパティは、add fs または add dataset によってゾーンにマウントされたファイルシステムには適用されません。

**Point**

ゾーン起動後に設定する場合は、zoneadm apply コマンドで起動中のノングローバルゾーンに適用可能です。

※ 詳細は、「[《参考》 ゾーンの設定の変更・削除](#)」を参照してください。

※ ノングローバルゾーンの再起動でも適用可能です。

**13) ゾーンの設定を検証します。**

```
zonecfg:zone01> verify
```

- ▶ 構成内容に矛盾がある場合は、メッセージが表示されます。  
例: ネットワークインターフェースが共有設定 (ip-type=shared) で anet を使用しようとした場合  
zonecfg:zone01> verify  
anet resources can only be specified if ip-type = exclusive  
zone01: Incompatible settings

**14) ゾーンの設定内容を確定します。**

```
zonecfg:zone01> commit
```

- ▶ 次の手順 15) で exit したときも、内部で設定内容の確定が実行されます。

**15) ゾーンの設定を終了します。**

通常のプロンプトに戻ります。

```
zonecfg:zone01> exit
```

**《参考》 便利な機能 (入力候補の表示とコマンドの補完)**

zonecfg コマンド実行中にプロンプト上で Tab キーを押すと、入力候補一覧が表示されます。

また、入力候補がある場合は、途中まで入力したコマンドを補完することもできます。

例: 「e」+ Tab キーを押した場合

```
zonecfg:zone01> e  
exit      export
```

## 2.3. ゾーンのインストール

### 《注意》

ゾーンのインストールには、リポジトリサーバが必要です。事前に用意してください。

### 《参考》 リポジトリサーバの構築方法

リポジトリサーバの構築方法については、以下を参照してください。

- 『Oracle Solaris 11 を使ってみよう(構築・運用手順書)』  
<https://www.fujitsu.com/jp/sparc-technical/document/solaris/#os>

### 1) インストール前にゾーン設定を確認します。

```
global01# zonecfg -z zone01 info -a
zonename: zone01
zonepath: /zones/zone01
brand: solaris
autoboot: false
autosutdown: shutdown
bootargs:
file-mac-profile:
pool: pool_1
limitpriv:
scheduling-class: FSS
ip-type: exclusive
hostid:
tenant:
fs-allowed:
anet:
    linkname: vnic0
    lower-link: net0
    allowed-address:
    configure-allowed-address: true
    defrouter:
    allowed-dhcp-cids:
    link-protection: mac-nospoof
    mac-address: auto
    auto-mac-address:
    mac-prefix:
    mac-slot:
    vlan-id:
    priority:
    rxrings:
    txrings:
    mtu:
    maxbw:
```

```

bwshare:
rxfanout:
vsi-typeid:
vsi-vers:
vsi-mgrid:
etsbw-lcl:
cos:
pkey:
linkmode:
evs:
vport:
ring-group: auto
autopush:

```

## 2) ゾーンの状態を表示します。

構成した zone01 の STATUS が「configured」と表示されることを確認します。

```

global01# zoneadm list -vc
ID NAME          STATUS    PATH                                BRAND  IP
0  global         running   /                                solaris shared
- zone01        configured /zones/zone01                  solaris excl

```

## 3) publisher を確認します。

publisher の STATUS が「online」と表示されることを確認します。

```

global01# pkg publisher
PUBLISHER          TYPE    STATUS P LOCATION
solaris            origin  online F http://localhost/

```

- ▶ 本書では、リポジトリサーバを localhost としています。

#### 4) ゾーンをインストールします。

##### i) ゾーンのインストールを行います。

ゾーンのインストールが問題なく完了すると、「Succeeded」と表示されます。

```
global01# zoneadm -z zone01 install
The following ZFS file system(s) have been created:
  zones/zone01
Progress being logged to /var/log/zones/zoneadm.20190513T083030Z.zone01.install
  Image: Preparing at /zones/zone01/root.

Install Log: /system/volatile/install.20431/install_log
AI Manifest: /tmp/manifest.xml.274Naa
SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
Zonename: zone01
Installation: Starting ...

    Creating IPS image
Startup linked: 1/1 done
    Installing packages from:
        solaris
        origin: http://localhost:11000/

DOWNLOAD                                PKGS          FILES      XFER (MB)   SPEED
Completed                              445/445      68840/68840  470.5/470.5  1.2M/s

PHASE                                ITEMS
Installing new actions                92791/92791
Updating package state database         Done
Updating package cache                 0/0
Updating image state                   Done
Creating fast lookup database          Done
Updating package cache                 1/1
Installation: Succeeded
done.

    Done: Installation completed in 673.812 seconds.

Next Steps: Boot the zone, then log into the zone console (zlogin -C)

            to complete the configuration process.

Log saved in non-global zone as
/zones/zone01/root/var/log/zones/zoneadm.20190513T083030Z.zone01.install
```

▶ zonepath で指定したフォルダは自動的に作成されます。

- ii) インストール完了後に、ゾーンの状態を確認します。

インストールが完了したゾーンの STATUS が「installed」と表示されることを確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS    | PATH          | BRAND   | IP     |
|----|--------|-----------|---------------|---------|--------|
| 0  | global | running   | /             | solaris | shared |
| -  | zone01 | installed | /zones/zone01 | solaris | excl   |

## 2.4. ゾーンの起動

- 1) ゾーンを起動します。

```
global01# zoneadm -z zone01 boot
```

▶ 数十秒、時間がかかります。

- 2) ゾーンの状態を確認します。

起動したゾーンの STATUS が「running」と表示されることを確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS  | PATH          | BRAND   | IP     |
|----|--------|---------|---------------|---------|--------|
| 0  | global | running | /             | solaris | shared |
| 1  | zone01 | running | /zones/zone01 | solaris | excl   |



## 2.5. 仮想 OS の初期設定

### 1) ゾーンのコンソールログインを実施します。

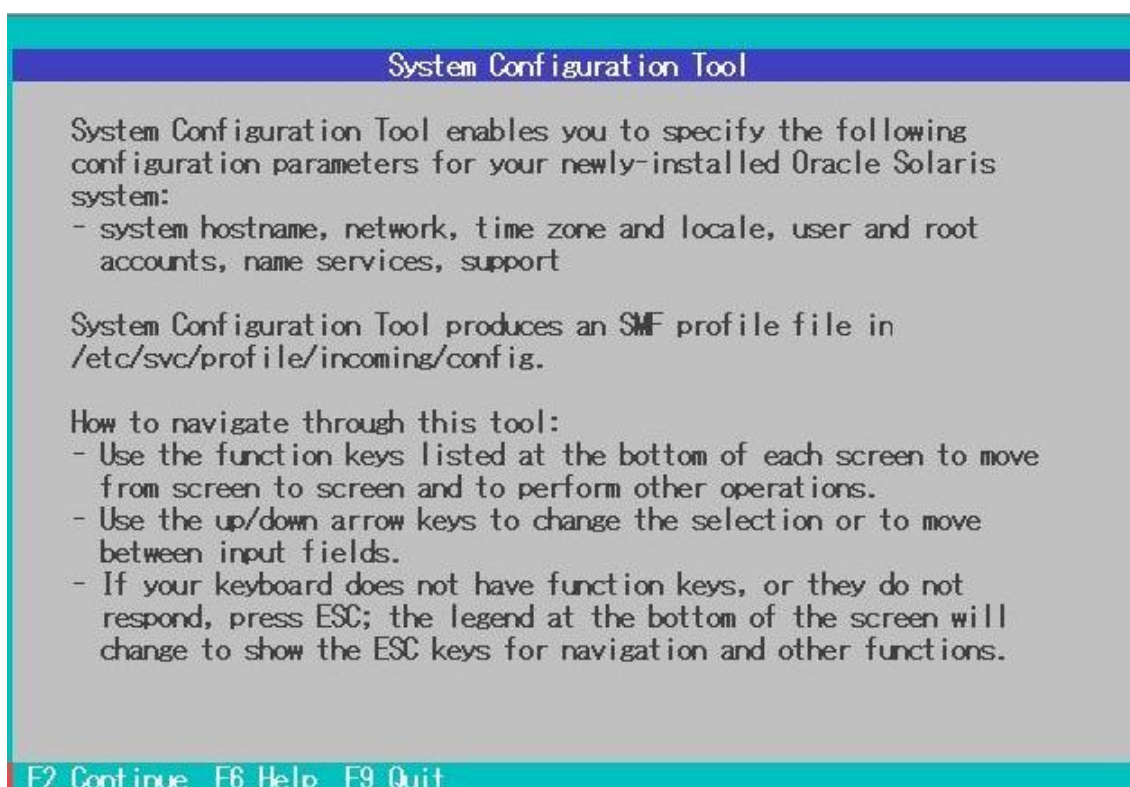
ゾーンのインストール完了後、コンソールログインを実施することで、ゾーンの初期設定が開始されます。

※ 初期設定が未実施の状態でもゾーンは起動しますが、必要なサービスが起動していない状態です。

※ ゾーンが起動している状態で実行してください。

```
global01# zlogin -G zone01
```

対話形式の System Configuration Tool が起動します。



- ▶ 初期画面が表示されるまで、数分かかります。
- ▶ ツール実行中にログアウトした場合などは、再ログインしても画面表示されない場合があります。その場合は、画面リフレッシュ ([Ctrl]+[L]キー) することで再表示されます。

### 2) ゾーンの初回セットアップを実行します。

#### 操作方法

入力が必要な画面では、矢印キー([↑]/[↓])でカーソル移動、スペースキーで選択、[Esc]+[2]キーまたは[F2]キーで次の画面へ移動します。

セットアップ後、初期設定で作成したユーザーでログインします。

## 設定値(例)

| 画面                           | 項目                   | 値                    |
|------------------------------|----------------------|----------------------|
| System Identity              | Computer Name        | zone01               |
| Network Configuration        | NIC Name             | vnic0                |
| Network Configuration Method | Configuration Method | Static               |
| Configure static IP address: | NIC:                 | vnic0/v4             |
|                              | IIP Address:         | 192.168.10.xx        |
|                              | Netmask:             | 255.255.255.0        |
|                              | Router:              | 192.168.10.xx        |
| DNS Name Service             | DNS                  | Do not configure DNS |
| Alternate Name Service       | Name Service         | None                 |
| Time Zone: Regions           | Regions              | Asia                 |
| Time Zone: Locations         | Locations            | Japan                |
| Time Zone                    | Time Zones           | Asia/Tokyo           |
| Locale: Language             | Language             | Japanese             |
| Locale: Territory            | Territory            | Japan (ja_JP.UTF-8)  |
| Users                        | Root password:       | xxxxxxxx             |
|                              | Confirm password:    | xxxxxxxx             |
|                              | Your real name:      | xxxxxx               |
|                              | User password:       | xxxxxxxx             |
|                              | Confirm password:    | xxxxxxxx             |

▶ Root password、User password は 8 文字以上、1 文字の英字と 1 つの数字または特殊文字を含める必要があります。

セットアップの完了後、console login プロンプトが表示されれば、初回セットアップは完了です。

```
zone01 console login:
```

▶ sendmail サービス関連のメッセージが表示されますが、問題ありません。

## 3) コンソールへのログインを確認します。

初回セットアップ時に設定した一般ユーザーでログインします。

```
zone01 console login: user01
Password:
```

#### 4) root ユーザーのログインを確認します。

su コマンドで、root ユーザーに変更できることを確認します。

※ 初回セットアップ時に一般ユーザーを作成した場合、root ユーザーは役割(role)として作成されるため、root ユーザーでは直接ログインできません。

```
zone01$ su -
Password:
```

#### 5) サービスの状態を確認します。

サービスの状態である STATE の部分に「legacy\_run」、「disabled」、または「online」以外の表示がないことを確認します。

※ Solaris 11.4 では、「svc:/network/tncpl:default」サービスが「uninitialized」となっていますが、問題ありません。

```
zone01# svcs -a
STATE          STIME      FMRI
legacy_run     18:05:56   lrc:/etc/rc2_d/S89PRESERVE
disabled       18:05:36   svc:/system/labeld:default
disabled       18:05:37   svc:/network/dns/client:default
disabled       18:05:37   svc:/network/firewall:default
~ (省略) ~
online         18:05:36   svc:/milestone/immutable-setup:default
online         18:05:36   svc:/network/socket-config:default
online         18:05:36   svc:/system/early-manifest-import:default
~ (省略) ~
uninitialized   18:05:40   svc:/network/tncpl:default
```

▶ STATE の表示に「\*」の印が付いている場合は、そのサービスが起動中であることを意味します。時間をおいて再確認してください。

#### 6) コンソール接続を閉じます。

「~」(チルダ)を入力し、「.」(ドット)を入力します。

「~」を入力した時点では画面に何も表示されません。「.」を入力した時点で「~.」と表示されます。

```
zone01# ~.
[Connection to zone 'zone01' console closed]
global01#
```

▶ 先に「~」のみ表示された場合は、[Enter]キーを押して、再実行してください。

## 《参考》ゾーンの設定の変更・削除

## Point

変更・削除したゾーンの設定を反映するには、ノングローバルゾーンの再起動が必要です。一部の設定項目については、zoneadm apply コマンドで動的に反映することが可能です（ノングローバルゾーンの再起動は不要）。

動的に設定可能な項目の詳細については、以下の Oracle 社のマニュアルを参照してください。

[https://docs.oracle.com/cd/E37838\\_01/html/E61040/ggqwk.html#scrolltoc](https://docs.oracle.com/cd/E37838_01/html/E61040/ggqwk.html#scrolltoc)

-Resource Types and Global Properties That Support Live Zone Reconfiguration

- リソースプールの関連付け設定変更(新しい設定が上書きされます)。

```
global01# zonecfg -z zone01
zonecfg:zone01> set pool=pool_default
zonecfg:zone01> commit
zonecfg:zone01> exit
global01# zoneadm -z zone01 apply
Checking: Clearing pool
Applying the changes
```

- ▶ zoneadm apply コマンドを実行すると、動作中のノングローバルゾーンに設定が反映されます。

- IP アドレスの設定(ip-type=shared の場合)

```
global01# zonecfg -z zone01
zonecfg:zone01> add net
zonecfg:zone01:net> set physical=net0
zonecfg:zone01:net> set address=192.168.10.XX/24
zonecfg:zone01:net> set defrouter=192.168.10.1
zonecfg:zone01:net> end
zonecfg:zone01> commit
zonecfg:zone01> exit
global01# zoneadm -z zone01 apply
Checking: Adding net physical=net0
Applying the changes
add net -ifp net0 default: gateway 192.168.10.1
```

- ▶ anet を設定済みの場合は、事前に anet の削除と ip-type= shared への変更が必要です。

- IP アドレスの変更(ip-type=shared の場合)

例:「192.168.10.XX/24」から「192.168.10.ZZ/24」に変更

```
global01# zonecfg -z zone01
zonecfg:zone01> select net address=192.168.10.XX/24
zonecfg:zone01:net> set address=192.168.10.ZZ/24
zonecfg:zone01:net> end
zonecfg:zone01> commit
zonecfg:zone01> exit
global01# zoneadm -z zone01 apply
Checking: Removing net physical=net0
Checking: Adding net physical=net0
Applying the changes
add net -ifp net0 default: gateway 192.168.10.1: entry exists
```

▶ ip-type=exclusive の場合は、ゾーン内で IP アドレスの変更を行います。

- 設定の削除

例: 共有ファイルシステムの設定を削除

```
global01# zonecfg -z zone01 "remove fs dir=/cdrom;"
global01# zoneadm -z zone01 apply
Checking: Unmounting fs dir=/cdrom
Applying the changes
```

- 設定の中止

例: 共有ファイルシステムの設定を中止

```
global01# zonecfg -z zone01
zonecfg:zone01> add fs
zonecfg:zone01:fs> set dir=/cdrom
zonecfg:zone01:fs> cancel
zonecfg:zone01>
```

### 《参考》ゾーンを使用停止(incomplete)にする

ゾーンを起動できない状態になります。incomplete の実行後は、ゾーンのアンインストールのみ可能です。ゾーンが停止した状態で実行してください。

```
global01# zoneadm -z zone01 mark incomplete
```

(1) リソースプール機能を停止します。

リソースプール構成を初期化します。

```
global01# pooladm -x
```

▶ リソースプールは、「pool\_default」および「pset\_default」のみの状態になります。

(2) pset\_default の設定を初期化します。

本書では、pset\_default の最低 CPU スレッド数を変更しているため、元の設定に戻します。  
なお、Oracle Solaris 11.4 では pooladm -x で初期化されるため、本手順は実施不要です。

```
global01# poolcfg -c 'modify pset pset_default (uint pset.min=1)'
```

(3) リソースプール構成を保存します。

初期化された構成をリソースプール構成ファイルに保存します。

```
global01# pooladm -s
```

(4) リソースプールサービスを停止します。

```
global01# svcadm disable svc:/system/pools:default
```

```
global01# svcadm disable svc:/system/pools/dynamic:default
```

(5) リソースプールの状態を確認します。

リソースプールサービスが停止していることを確認します。

```
global01# svcs -a | grep pools
disabled    10:53:34 svc:/system/pools:default
disabled    10:53:36 svc:/system/pools/dynamic:default
```

### 《参考》ゾーンの削除

(1) ゾーンをアンインストールします。

(i) ゾーンの状態を確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS  | PATH          | BRAND   | IP     |
|----|--------|---------|---------------|---------|--------|
| 0  | global | running | /             | solaris | shared |
| 1  | zone01 | running | /zones/zone01 | solaris | excl   |

(ii) ゾーンを停止します。

アンインストールするゾーンが起動している場合に停止します。

```
global01# zoneadm -z zone01 shutdown
```

(iii) ゾーンの状態を確認します。

ゾーンが停止している (STATUS に installed が表示されている) ことを確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS    | PATH          | BRAND   | IP     |
|----|--------|-----------|---------------|---------|--------|
| 0  | global | running   | /             | solaris | shared |
| -  | zone01 | installed | /zones/zone01 | solaris | shared |

(iv) ゾーンをアンインストールします。

確認メッセージが表示されるので、「y」を入力します。

```
global01# zoneadm -z zone01 uninstall
Are you sure you want to uninstall zone zone01 (y/[n])? y
```

▶ -F オプションを付けると強制的に処理を実行します。

(v) ゾーンの状態確認

ゾーンがアンインストールされた (STATUS に configured が表示されている) ことを確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS     | PATH          | BRAND   | IP     |
|----|--------|------------|---------------|---------|--------|
| 0  | global | running    | /             | solaris | shared |
| -  | zone01 | configured | /zones/zone01 | solaris | shared |

(2) ゾーンを削除します。

(i) ゾーンの構成情報を削除します。

確認メッセージが表示されるので、「y」を入力します。

```
global01# zonecfg -z zone01 delete
Are you sure you want to delete zone zone01 (y/[n])? y
```

(ii) ゾーンの状態を確認します。

ゾーンが削除されたことを確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS  | PATH | BRAND   | IP     |
|----|--------|---------|------|---------|--------|
| 0  | global | running | /    | solaris | shared |

リソースプールの初期化や停止が必要な場合は、「[《参考》リソースプールの初期化と停止](#)」を参照してください。

### 3. ゾーン(仮想 Solaris 環境)の基本操作

#### 3.1. ゾーンの状態の確認

##### 1) ゾーンの状態を表示します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS  | PATH          | BRAND   | IP     |
|----|--------|---------|---------------|---------|--------|
| 0  | global | running | /             | solaris | shared |
| 4  | zone01 | running | /zones/zone01 | solaris | excl   |

- ▶ NAME: ゾーン名
- STATUS: 実行状態
- PATH: ゾーンパス
- BRAND: ブランドタイプ
- IP: IP タイプ

#### 《参考》再起動実行中のゾーンのステータスの確認

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS  | PATH          | BRAND   | IP     |
|----|--------|---------|---------------|---------|--------|
| 0  | global | running | /             | solaris | shared |
| 5  | zone01 | ready   | /zones/zone01 | solaris | excl   |

- ▶ 状態が「running」になるまで数十秒かかります。



## 3.2. ゾーンの起動／停止

### 3.2.1. ゾーンの起動・再起動・停止操作（通常運用時）

通常運用時のゾーンの起動、再起動、および停止操作を以下に示します。

- ゾーンの再起動

【グローバルゾーン上での実行】

```
global01# zlogin zone01 'shutdown -y -g0 -i6'
```

【ノングローバルゾーン上での実行】

```
global01# zlogin zone01
[Connected to zone 'zone01' pts/1]
Last login: Thu Apr 25 17:54:33 2019 on pts/1
Oracle Corporation      SunOS 5.11      11.4      January 2019
zone01#
zone01# shutdown -y -g0 -i6

Shutdown started.      Thursday, April 25, 2019 at  5:55:50 PM JST

Changing to init state 6 - please wait
Broadcast Message from root (pts/1) on zone01 Thu Apr 25 17:55:50...
THE SYSTEM zone01 IS BEING SHUT DOWN NOW !!!
Log off now or risk your files being damaged

zone01#
[Connection to zone 'zone01' pts/1 closed]
global01#
```

- ゾーンの停止

【グローバルゾーン上での実行】

```
global01# zlogin zone01 'shutdown -y -g0 -i5'
```

または

```
global01# zoneadm -z zone01 shutdown
```

▶ Oracle Solaris 11 から、zoneadm の shutdown サブコマンドが使用できます。

## 【ノングローバルゾーン上での実行】

```
global01# zlogin zone01
[Connected to zone 'zone01' pts/1]
Last login: Thu Apr 25 17:55:07 2019 on pts/1
Oracle Corporation      SunOS 5.11      11.4      January 2019
zone01#
zone01# shutdown -y -g0 -i5

Shutdown started.      Thursday, April 25, 2019 at  6:02:45 PM JST

Changing to init state 5 - please wait
Broadcast Message from root (pts/1) on zone01 Thu Apr 25 18:02:45...
THE SYSTEM zone01 IS BEING SHUT DOWN NOW !!!
Log off now or risk your files being damaged

zone01#
[Connection to zone 'zone01' pts/1 closed]
global01#
```

## ● ゾーンのシングルユーザーモード起動

```
global01# zoneadm -z zone01 boot -s
```

## ● ゾーンの起動

```
global01# zoneadm -z zone01 boot
```

### 3.2.2. ゾーンの起動・再起動・停止操作(緊急時)

緊急時のゾーンの起動、再起動、および停止操作を以下に示します。

#### 《注意》

reboot(再起動)や halt(停止)は、ゾーン内のプロセスの状態に関わらず実行されます。アプリケーションなどがゾーン上で実行されている場合は、通常運用時の操作を行います。  
本書では、便宜的に上記コマンドを主に利用していますが、実運用では注意してください。

- ゾーンの再起動

```
global01# zoneadm -z zone01 reboot
```

- ゾーンの停止

```
global01# zoneadm -z zone01 halt
```

- ゾーンの待機

```
global01# zoneadm -z zone01 ready
```

- ゾーンの起動

```
global01# zoneadm -z zone01 boot
```

### 3.3. ゾーンへのログイン

- コンソールモード

ゾーン停止中でも接続できます。同時に 1 ユーザーのみ接続可能です。ゾーンを再起動しても切断されることはありません。

【書式】 zlogin -C <ゾーン名>

```
global01# zlogin -C zone01
[Connected to zone 'zone01' console]
```

▶ 「exit」または「~」でゾーンからログアウトできます。

- インタラクティブモード(対話型)

ゾーン起動中のみ接続できます。同時に複数ユーザーが接続可能です。ログイン時のパスワード認証はありません。

【書式】 zlogin <ゾーン名>

```
global01# zlogin zone01
[Connected to zone 'zone01' pts/1]
Last login: Thu Apr 25 18:02:31 2019 on pts/1
Oracle Corporation      SunOS 5.11      11.4      January 2019
```

▶ 「exit」または「~」でゾーンからログアウトできます。

### 3.4. ゾーンへのコマンド実行

ノンインタラクティブモード(非対話型)でのゾーン操作は、以下のように実行します。

```
global01# zlogin zone01 'cat /etc/hosts'
#
# Copyright 2009 Sun Microsystems, Inc. All rights reserved.
# Use is subject to license terms.
#
# Internet host table
#
::1          localhost
127.0.0.1    localhost loghost
192.168.10.XX zone01
global01#
```

- ▶ ゾーン上で実行したコマンドの結果を表示します。
- ▶ -l オプション+ユーザー名で、実行ユーザーを指定することができます。  
global01# zlogin -l user01 zone01 'cat /etc/hosts'

### 3.5. ゾーン情報の確認

#### 3.5.1. グローバルゾーンでのゾーン情報の確認

- 動作しているプロセスを表示

```
global01# ps -ef
  UID  PID  PPID  C   STIME TTY          TIME CMD
  root    0    0    0   Dec 25 ?        0:12 sched
  root    5    0    0   Dec 25 ?        6:18 zpool-rpool
  root    6    0    0   Dec 25 ?        0:06 kmem_task
~ (省略) ~
```

- 動作しているプロセス数を表示

```
global01# ps -ef | wc -l
131
```

- プロセスが動作しているゾーン名を表示して確認

```
global01# ps -efZ
  ZONE  UID  PID  PPID  C   STIME TTY          TIME CMD
  global root    0    0    0   Dec 25 ?        0:12 sched
  global root    5    0    0   Dec 25 ?        6:18 zpool-rpool
~ (省略) ~
  zone01 root 16602    1    0 09:49:35 ?        0:00 /usr/lib/labeld
~ (省略) ~
```

- zone01 で動作しているプロセスのみを表示

```
global01# ps -efZ | grep zone01
  zone01 root 16248    1    0 09:49:32 ?        0:00 /lib/inet/in.ndpd
  zone01 root 16781    1    0 09:49:39 ?        0:00 /usr/lib/fm/fmd/fmd
~ (省略) ~
```

- 動作しているプロセスのスケジューリングクラスを表示して確認

```
global01# ps -efc
  UID  PID  PPID  CLS PRI  STIME TTY          TIME CMD
  root    0    0   SYS  96   Dec 25 ?        0:12 sched
  root    5    0   SDC  99   Dec 25 ?        6:18 zpool-rpool
  root    6    0   SDC  99   Dec 25 ?        0:06 kmem_task
  root    1    0   FSS   1   Dec 25 ?        0:00 /usr/sbin/init
~ (省略) ~
```

- 指定したプロセスのみを表示

```
global01# ps -efc | grep /sbin/init
  root    1    0  FSS   1   Dec 25 ?        0:00 /usr/sbin/init
  root 16128 16064  FSS   1 09:49:26 ?        0:00 /usr/sbin/init
  root 17069  9136  FSS  59 10:43:34 console 0:00 grep /sbin/init
```

### 3.5.2. ノングローバルゾーンでのゾーン情報の確認

- インタラクティブモード(対話型)でログイン

```
global01# zlogin zone01
```

- 動作しているプロセスを表示

```
zone01# ps -ef
root 16248 16064 0 09:49:32 ? 0:00 /lib/inet/in.ndpd
root 16781 16064 0 09:49:39 ? 0:00 /usr/lib/fm/fmd/fmd
root 17091 16064 0 10:52:47 pts/2 0:00 /usr/bin/login -z global -f root
~ (省略) ~
```

- 動作しているプロセス数を表示

```
zone01# ps -ef | wc -l
44
```

- プロセスが動作しているゾーン名を表示して確認

```
zone01# ps -efZ
ZONE      UID    PID  PPID  C   STIME TTY      TIME CMD
zone01    root  16248 16064 0   09:49:32 ?      0:00 /lib/inet/in.ndpd
zone01    root  16781 16064 0   09:49:39 ?      0:00 /usr/lib/fm/fmd/fmd
~ (省略) ~
```

- zone01 で動作しているプロセスのみを表示

```
zone01# ps -efZ | grep zone01
ZONE      UID    PID  PPID  C   STIME TTY      TIME CMD
zone01    root  16248 16064 0   09:49:32 ?      0:00 /lib/inet/in.ndpd
zone01    root  16781 16064 0   09:49:39 ?      0:00 /usr/lib/fm/fmd/fmd
~ (省略) ~
```

▶ 結果は同じです。

- 動作しているプロセスのスケジューリングクラスを表示して確認

```
zone01# ps -efc
UID    PID  PPID  CLS PRI   STIME TTY      TIME CMD
root  16248 16064  FSS  1 09:49:32 ?      0:00 /lib/inet/in.ndpd
root  16781 16064  FSS 45 09:49:39 ?      0:00 /usr/lib/fm/fmd/fmd
root  17091 16064  FSS  1 10:52:47 pts/2 0:00 /usr/bin/login -z global -f root
root
~ (省略) ~
```

- 指定したプロセスのみを表示

```
zone01# ps -efc | grep /sbin/init
root  16128 16064  FSS  1 09:49:26 ?      0:00 /usr/sbin/init
root  17135 17092  FSS  1 10:54:12 pts/2 0:00 grep /sbin/init
```

- インタラクティブモードの接続を閉じる

```
zone01# exit
logout

[Connection to zone 'zone01' pts/2 closed]
global01#
```

### 3.6. ゾーンの構成情報の変更

既存のゾーンに対して構成情報を変更し、設定を反映します。

例として、ここでは以下のプロパティを設定します。

- CPU の占有設定 (dedicated-cpu)
- メモリ制限設定 (capped-memory)

なお、設定するノングローバルゾーンに対してリソースプールを割り当てている場合は、CPU の占有設定ができません。

このため、リソースプールの割り当てを解除してから設定を行う必要があります。

#### 3.6.1. ゾーンの停止

- 1) 現在の状態を確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS  | PATH          | BRAND   | IP     |
|----|--------|---------|---------------|---------|--------|
| 0  | global | running | /             | solaris | shared |
| 1  | zone01 | running | /zones/zone01 | solaris | excl   |

- 2) ゾーンを停止します。

```
global01# zoneadm -z zone01 shutdown
```

- 3) 状態を確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS    | PATH          | BRAND   | IP     |
|----|--------|-----------|---------------|---------|--------|
| 0  | global | running   | /             | solaris | shared |
| -  | zone01 | installed | /zones/zone01 | solaris | excl   |

### 3.6.2. ゾーン構成情報の変更

#### 1) 現在の設定値を確認します。

「pool\_1」というリソースプールが割り当てられています。

```
global01# zonecfg -z zone01
zonecfg:zone01> info -a
zonename: zone01
zonpath: /zones/zone01
brand: solaris
autoboot: false
autoshutdown: shutdown
bootargs:
file-mac-profile:
pool: pool_1
limitpriv:
scheduling-class: FSS
ip-type: exclusive
hostid:
tenant:
fs-allowed:
~ (省略) ~
```

#### 2) リソースプールの設定を解除します。

```
zonecfg:zone01> clear pool
```

#### 3) CPU の占有設定を実施します。

ここでは、2 CPU を占有する設定にしています。

```
zonecfg:zone01> add dedicated-cpu
zonecfg:zone01:dedicated-cpu> set ncpus=2
zonecfg:zone01:dedicated-cpu> end
```

#### 4) メモリ制限設定を実施します。

ここでは、メモリ制限を 1 GB に設定しています。

```
zonecfg:zone01> add capped-memory
zonecfg:zone01:capped-memory> set physical=1G
zonecfg:zone01:capped-memory> end
```



## 5) ゾーンの設定を確認します。

```
zonecfg:zone01> info -a
zonename: zone01
zonepath: /zones/zone01
brand: solaris
autoboot: false
autosutdown: shutdown
bootargs:
file-mac-profile:
pool:
limitpriv:
scheduling-class: FSS
ip-type: exclusive
hostid:
tenant:
fs-allowed:
~ (省略) ~
dedicated-cpu:
    ncpus: 2
    cpus:
    cores:
    sockets:
capped-memory:
    physical: 1G
```

## 6) ゾーンの設定内容を確定します。

```
zonecfg:zone01> commit
```

## 7) ゾーンの設定を終了します。

```
zonecfg:zone01> exit
```

## 3.6.3. ゾーンの起動

## 1) ゾーンを起動します。

```
global01# zoneadm -z zone01 boot
```

▶ ゾーンに割り当てる CPU 空きリソースがない場合は、起動に失敗します。

## 2) 状態を確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS  | PATH          | BRAND   | IP     |
|----|--------|---------|---------------|---------|--------|
| 0  | global | running | /             | solaris | shared |
| 12 | zone01 | running | /zones/zone01 | solaris | excl   |

### 3) CPU の占有設定を確認します。

ゾーン起動時に自動で SYSzone\_zone01 というリソースプールが作成され、zone01 に 2 CPU が割り当てられていることを確認します。

```
global# poolstat
```

| id | pool           | pset |      |      |
|----|----------------|------|------|------|
|    |                | size | used | load |
| 1  | pool_1         | 4    | 0.00 | 0.00 |
| 0  | pool_default   | 42   | 0.00 | 0.17 |
| 2  | SYSzone_zone01 | 2    | 0.00 | 1.60 |

▶ ゾーン停止時に自動で削除されます。

### 4) メモリの制限設定を確認します。

zone01 に 1024 MB (1 GB) が割り当てられていることを確認します。

```
global01# zlogin zone01 prtconf
prtconf: devinfo facility not available
System Configuration: Oracle Corporation sun4v
Memory size: 1024 Megabytes
System Peripherals (Software Nodes):
```

#### 《参考》 zoneadm apply コマンドを使用した場合

「[3.6.2. ゾーン構成情報の変更](#)」を実施後、zoneadm apply コマンドで動的に反映できます。

ノングローバルゾーンを再起動する必要がないため、「[3.6.1. ゾーンの停止](#)」と「[3.6.3. ゾーンの起動](#)」が不要となります。

「3.6.2. ゾーン構成情報の変更」の[手順 5](#)のあとに、以下のコマンドを実行します。

```
global01# zoneadm -z zone01 apply
Checking: Adding capped-memory physical=1G
Checking: Adding dedicated-cpu
Applying the changes
```

▶ ノングローバルゾーンを停止せずに設定の反映が可能です。

#### 3.6.4. 設定したプロパティの削除

##### 1) ゾーンを停止します。

```
global01# zoneadm -z zone01 shutdown
global01# zoneadm list -vc
```

| ID | NAME   | STATUS    | PATH          | BRAND   | IP     |
|----|--------|-----------|---------------|---------|--------|
| 0  | global | running   | /             | solaris | shared |
| -  | zone01 | installed | /zones/zone01 | solaris | excl   |

- 2) 設定したプロパティ(`dedicated-cpu` と `capped-memory`)を削除し、リソースプールに「`pool_1`」を設定します。

```
global01# zonecfg -z zone01
zonecfg:zone01> info -a
zonename: zone01
zonepath: /zones/zone01
brand: solaris
autoboot: false
autoshutdown: shutdown
bootargs:
file-mac-profile:
pool: 
limitpriv:
scheduling-class: FSS
ip-type: exclusive
hostid:
tenant:
fs-allowed:
~ (省略) ~
dedicated-cpu:
    ncpus: 2
    cpus:
    cores:
    sockets:
capped-memory:
    physical: 1G
zonecfg:zone01> remove dedicated-cpu
zonecfg:zone01> remove capped-memory
zonecfg:zone01> set pool=pool_1
zonecfg:zone01> info -a
zonename: zone01
zonepath: /zones/zone01
brand: solaris
autoboot: false
autoshutdown: shutdown
bootargs:
file-mac-profile:
pool: pool_1
limitpriv:
scheduling-class: FSS
ip-type: exclusive
hostid:
tenant:
fs-allowed:
~ (省略) ~
zonecfg:zone01> commit
zonecfg:zone01> exit
```

## 3) ゾーンを起動します。

```
global01# zoneadm -z zone01 boot
global01# zoneadm list -vc
```

| ID | NAME   | STATUS  | PATH          | BRAND   | IP     |
|----|--------|---------|---------------|---------|--------|
| 0  | global | running | /             | solaris | shared |
| 12 | zone01 | running | /zones/zone01 | solaris | excl   |

## 4) CPU の占有設定が解除されていることを確認します。

リソースプール「SYSzone\_zone01」が削除されていることを確認します。

```
global01# poolstat
```

| id | pool         | size | used | load |
|----|--------------|------|------|------|
| 1  | pool_1       | 4    | 0.00 | 0.30 |
| 0  | pool_default | 12   | 0.00 | 0.25 |

▶ ゾーン停止時に自動で削除されます。

## 5) メモリの制限設定が解除されていることを確認します。

zone01 に 16384 MB (16 GB) が割り当てられていることを確認します。

```
global01# zlogin zone01 prtconf
prtconf: devinfo facility not available
System Configuration: Oracle Corporation sun4v
Memory size: 16384 Megabytes
System Peripherals (Software Nodes):
```

### 3.7. ゾーンの統計情報の確認

#### 1) 統計情報を確認します。

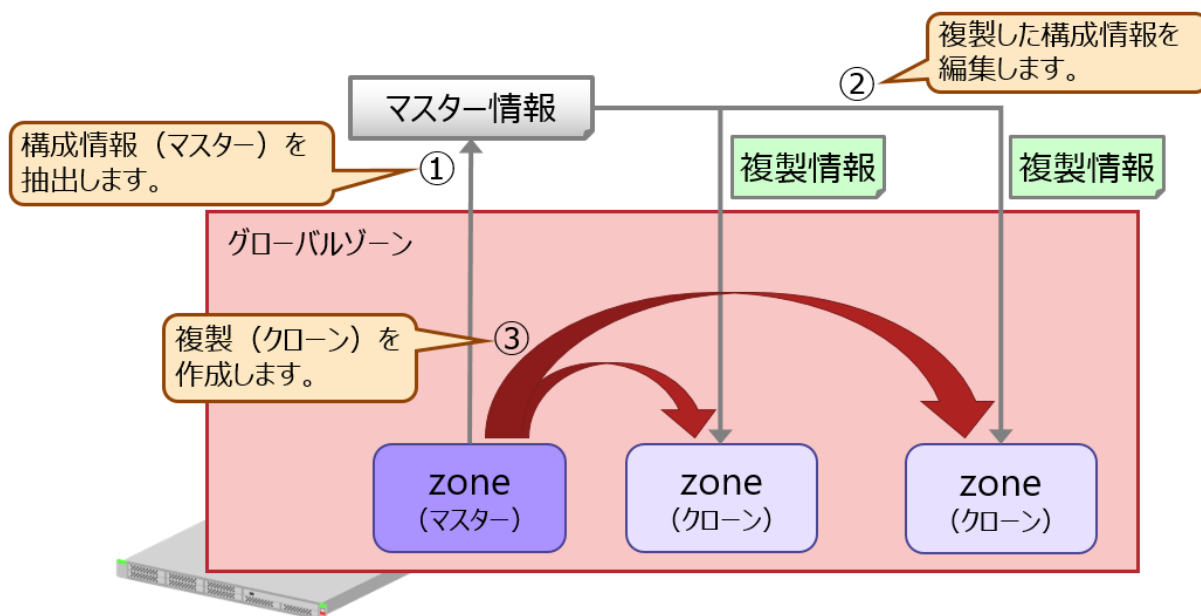
```
global01# zonestat 5
Collecting data for first interval...
Interval: 1, Duration: 0:00:05
SUMMARY          Cpus/Online: 48/48    PhysMem: 125G    VirtMem: 129G
                  -----CPU-----  --PhysMem--  --VirtMem--  --PhysNet--
                ZONE  USED %PART  STLN %STLN  USED %USED  USED %USED  PBYTE %PUSE
[total]         0.02 0.04%   0.00 0.00% 9355M 7.27% 10.7G 8.31%   0 0.00%
[system]        0.00 0.00%   0.00 0.00% 8341M 6.49% 9903M 7.46%   -  -
global          0.02 0.04%   -  -    702M 0.54%  803M 0.60%   0 0.00%
zone01          0.00 0.03%   -  -    310M 0.24%  314M 0.23%   0 0.00%
```

- ▶ 5 秒間隔でゾーンの統計情報を出力します。
- ▶ 停止する場合は、[Ctrl]+[C]キーを押します。

zonestat コマンドの出力フィールドは、以下のとおりです。

- CPU : CPU 使用量
- PhysMem : 物理メモリ使用量
- VirtMem : 仮想メモリ使用量
- PhysNet : ネットワーク使用量

## 4. ゾーン(仮想 Solaris 環境)の複製



### 4.1. 構成情報の作成

#### 1) 構成ファイルを作成して編集します。

- i) ゾーンを停止します。

```
global01# zoneadm -z zone01 shutdown
```

- ii) ゾーンの状態を表示します。

ゾーンが停止している (STATUS が「installed」になっている) ことを確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS    | PATH          | BRAND   | IP     |
|----|--------|-----------|---------------|---------|--------|
| 0  | global | running   | /             | solaris | shared |
| -  | zone01 | installed | /zones/zone01 | solaris | excl   |

- iii) ゾーンの構成ファイルを複製します。

```
global01# zonecfg -z zone01 export -f /zones/zone01_master_file
```

- iv) 編集前の/zones/zone01\_master\_fileを確認します。

```
global01# cat /zones/zone01_master_file
create -b
set brand=solaris
set zonepath=/zones/zone01
set pool=pool_1
set scheduling-class=FSS
add anet
set linkname=vnic0
set lower-link=net0
set configure-allowed-address=true
end
```

- v) 複製した構成ファイルを編集します。

```
global01# vi /zones/zone01_master_file
```

以下のように編集します。

zonepath は複製元の構成と重複するため、変更する必要があります。

```
global01# cat /zones/zone01_master_file
create -b
set brand=solaris
set zonepath=/zones/zones_clone/zone02
set pool=pool_1
set scheduling-class=FSS
add anet
set linkname=vnic0
set lower-link=net0
set configure-allowed-address=true
end
```

- 2) 編集した複製ファイルを使って新規ゾーンを構成します。

編集した zone01 の複製ファイルを使用し、新規ゾーン「zone02」を作成します。

```
global01# zonecfg -z zone02 -f /zones/zone01_master_file
```

- 3) ゾーンの状態を確認します。

zone02 が新規に構成されていることを確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS     | PATH                      | BRAND   | IP     |
|----|--------|------------|---------------------------|---------|--------|
| 0  | global | running    | /                         | solaris | shared |
| -  | zone01 | installed  | /zones/zone01             | solaris | excl   |
| -  | zone02 | configured | /zones/zones_clone/zone02 | solaris | excl   |

## 4.2. クローン(複製)の作成

### 1) zone01 の複製を zone02 に作成します。

```
global01# zoneadm -z zone02 clone zone01
The following ZFS file system(s) have been created:
  zones/zones_clone
  zones/zones_clone/zone02
Progress being logged to /var/log/zones/zoneadm.20190107T050310Z.zone02.clone
Log saved in non-global zone as
/zones/zones_clone/zone02/root/var/log/zones/zoneadm.20190107T050310Z.zone02.clone
```

▶ zonepath で指定したフォルダは自動的に作成されます。

### 2) ゾーンの状態を確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS    | PATH                      | BRAND   | IP     |
|----|--------|-----------|---------------------------|---------|--------|
| 0  | global | running   | /                         | solaris | shared |
| -  | zone01 | installed | /zones/zone01             | solaris | excl   |
| -  | zone02 | installed | /zones/zones_clone/zone02 | solaris | excl   |

### 3) zone02 を起動します。

```
global01# zoneadm -z zone02 boot
```

### 4) ゾーンの状態を確認確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS    | PATH                      | BRAND   | IP     |
|----|--------|-----------|---------------------------|---------|--------|
| 0  | global | running   | /                         | solaris | shared |
| 2  | zone02 | running   | /zones/zones_clone/zone02 | solaris | excl   |
| -  | zone01 | installed | /zones/zone01             | solaris | excl   |

## 4.3. クローン(複製)の初期設定

### 1) ゾーンのコソールログイン(初期設定)を実施します。

ゾーンのインストールが完了したあと、コソールログインを実施することで、ゾーンの初期設定が開始します。

※ 初期設定が未実施の状態でもゾーンは起動しますが、必要なサービスが起動していない状態です。

※ ゾーンが起動している状態で実行してください。

```
global01# zlogin -C zone02
```

### 2) ゾーンの初回セットアップを実施します。

zone01 の初回セットアップ時と同様に、System Configuration Tool を使用します。

System Configuration Tool については、「[2.5. 仮想 OS の初期設定](#)」を参照してください。

※ ツール実行中にログアウトした場合などは、再ログインしても画面表示されない場合があります。



セットアップの完了後、console login プロンプトが表示されれば、初回セットアップは完了です。

```
zone02 console login:
```

▶ sendmail サービス関連のメッセージが表示されますが、問題ありません。

### 3) コンソールへのログインを確認します。

```
zone02 console login: user01
Password:
```

### 4) root ユーザーのログインを確認します。

```
zone02$ su -
Password:
```

### 5) サービスの状態を確認します。

サービスの状態である STATE の部分に「legacy\_run」、「disabled」、または「online」以外の表示がないことを確認します。

※ Solaris 11.4 では、「svc:/network/tnctld:default」サービスが「uninitialized」となっていますが、問題ありません。

```
zone02# svcs -a
STATE      STIME      FMRI
legacy_run 14:20:45 lrc:/etc/rc2_d/S89PRESERVE
disabled   14:05:14  svc:/system/labeld:default
disabled   14:05:16  svc:/network/dns/client:default
disabled   14:05:16  svc:/network/firewall:default
~ (省略) ~
online     14:05:14  svc:/system/svc/restarter:default
online     14:05:15  svc:/milestone/immutable-setup:default
online     14:05:16  svc:/network/datalink-management:default
~ (省略) ~
uninitialized 14:05:19  svc:/network/tnctld:default
```

▶ STATE の表示に、「\*」の印が付いている場合は、そのサービスが起動中であることを意味します。時間をおいて再確認してください。

### 6) コンソール接続を閉じます。

「~」(チルダ)を入力し、「.」(ドット)を入力します。

```
zone02# ~.
[Connection to zone 'zone02' console closed]
global01#
```

## 5. ゾーン(仮想 Solaris 環境)の移動

既存のゾーンを別のサーバへ移動し、同じゾーン環境を復元する方法を説明します。

本書では、以下の2つの方法でゾーンを移動します。

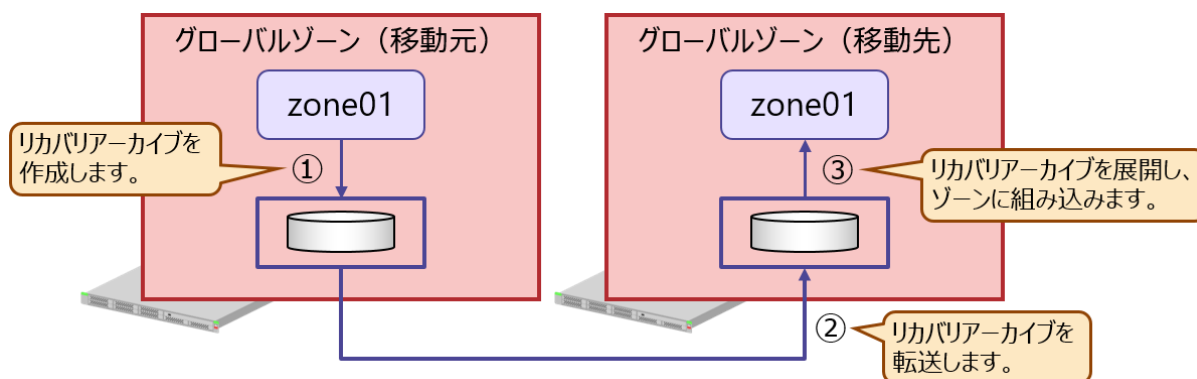
- 統合アーカイブを使用したゾーンの移動
- ZFS アーカイブを使用したゾーンの移動

### 《注意》

移動元サーバ(global01)でリソースプール「pool\_1」を作成して zone01 に割り当てていた場合は、移動先サーバ(global02)で事前にリソースプール「pool\_1」を作成しておく必要があります。

本書は、移動元サーバと移動先サーバの SRU 版数を同一にしています。

### 5.1. 統合アーカイブを使用したゾーンの移動



### 《注意》

統合アーカイブを使用したゾーンの移動は、Solaris 11.2 以降の環境が対象となります。

Solaris 11.1 以前の環境の場合は、「[5.2. ZFS アーカイブを使用したゾーンの移動](#)」を実施してください。

### 5.1.1. リカバリーアーカイブの作成と転送

#### 1) ゾーンのリカバリーアーカイブを作成します。

【書式】`archiveadm create -r -z <ゾーン名> <アーカイブ名.uar>`

```
global01# archiveadm create -r -z zone01 /zones/zone01-archive.uar

Logging to /system/volatile/archive_log.9125

0% : Beginning archive creation: /zones/zone01-archive.uar
6% : Executing dataset discovery...
10% : Dataset discovery complete
10% : Executing staging capacity check...
11% : Staging capacity check complete
15% : Creating zone media: UnifiedArchive [0bff29f9-0b88-4441-a190-da1e71702c64]
53% : CreateZoneMedia: UnifiedArchive [0bff29f9-0b88-4441-a190-da1e71702c64] complete
55% : Preparing archive image...
73% : Archive image preparation complete
75% : Beginning archive stream creation...
93% : Archive stream creation complete
93% : Beginning archive descriptor creation...
94% : Archive descriptor creation complete
95% : Beginning final archive assembly...
100% : Archive assembly complete
```

▶ リカバリーアーカイブの作成には、数分～数十分かかります。

## 2) 移動先サーバ(global02)へリカバリーアーカイブを転送します。

ここでは例として、移動先サーバ(global02)の/export/home/user01 にリカバリーアーカイブを格納します。

```
global01# ftp 192.168.10.ZZ
Connected to 192.168.10.ZZ
220 ::ffff:192.168.10.ZZ FTP server ready
Name (192.168.10.ZZ:user01): user01      ← 一般ユーザーでログイン
331 Password required for user01
Password: user01                        ← ログインユーザーのパスワードを入力（非表示）
230 User user01 logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> bin                                ← バイナリ転送モードを指定
200 Type set to I
ftp> cd /export/home/user01             ← 転送先ディレクトリの移動
250 CWD command successful
ftp> lcd /zones                         ← 転送元ディレクトリの移動
Local directory now /zones
ftp> !pwd                               ← 転送元ディレクトリの確認
/zones
ftp> put zone01-archive.uar             ← リカバリーアーカイブの転送
200 PORT command successful              (本書では/export/home/user01 配下に転送)
150 Opening BINARY mode data connection for zone01-archive.uar
226 Transfer complete
local: zone01-archive.uar remote: zone01-archive.uar
1796003840 bytes sent in 16 seconds (112219.82 Kbytes/s)
ftp> bye                                ← ftp の終了
221 Goodbye.
```

### 5.1.2. リカバリーアーカイブの展開とゾーンのインストール

#### 1) 移動元サーバ(global01)で、ゾーンを停止します。

```
global01# zoneadm -z zone01 shutdown
```

#### 2) ゾーンの状態を確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS    | PATH          | BRAND   | IP     |
|----|--------|-----------|---------------|---------|--------|
| 0  | global | running   | /             | solaris | shared |
| -  | zone01 | installed | /zones/zone01 | solaris | excl   |

#### 3) 移動先サーバ(global02)でリカバリーアーカイブを使用し、ゾーンを構成します。

```
global02# zonecfg -z zone01 create -a /export/home/user01/zone01-archive.uar
```

## 4) ゾーンの状態を確認します。

```
global02# zoneadm list -vc
ID NAME          STATUS      PATH                                BRAND  IP
0 global         running     /                                solaris shared
- zone01         configured  /zones/zone01                    solaris excl
```

## 5) リカバリーアーカイブからゾーンをインストールします。

ゾーンのインストールが成功すると、「Attach Succeeded.」と表示されます。

```
global02# zoneadm -z zone01 install -a /export/home/user01/zone01-archive.uar
The following ZFS file system(s) have been created:
  rpool/zones
  rpool/zones/zone01
Progress being logged to /var/log/zones/zoneadm.20190108T063636Z.zone01.install
Image: Preparing at /zones/zone01/root.

Install Log: /system/volatile/install.2224/install_log
AI Manifest: /tmp/manifest.zone01.GqG3Yc.xml
Zonename: zone01
Installation: Starting ...

Commencing transfer of stream: fe94ba52-ad53-454f-8788-2954d27ee993-0.zfs to
rpool/zones/zone01/rpool
Completed transfer of stream: 'fe94ba52-ad53-454f-8788-2954d27ee993-0.zfs'
from file:///export/home/user01/zone01-archive.uar
Archive transfer completed
Installation: Succeeded
Updating image format
Image format already current.
Updating non-global zone: Linking to image /.
Updating non-global zone: Syncing packages.
No updates necessary for this image. (zone:zone01)
Updating non-global zone: Zone updated.
Result: Attach Succeeded.
done.

Done: Installation completed in 123.714 seconds.

Next Steps: Boot the zone, then log into the zone console (zlogin -C)

to complete the configuration process.

Log saved in non-global zone as
/zones/zone01/root/var/log/zones/zoneadm.20190108T063636Z.zone01.install
```

▶ ゾーンのインストールには、数分かかります。

## 6) ゾーンの状態を確認します。

```
global02# zoneadm list -vc
```

| ID | NAME   | STATUS    | PATH          | BRAND   | IP     |
|----|--------|-----------|---------------|---------|--------|
| 0  | global | running   | /             | solaris | shared |
| -  | zone01 | installed | /zones/zone01 | solaris | excl   |

## 7) リソースプールを確認します。

```
global02# poolstat -r all
```

| id | pool         | type | rid | rset         | min | max | size | used | load |
|----|--------------|------|-----|--------------|-----|-----|------|------|------|
| 1  | pool_1       | pset | 1   | pset_1       | 4   | 4   | 4    | 0.00 | 0.00 |
| 0  | pool_default | pset | -1  | pset_default | 4   | 66K | 4    | 0.00 | 0.59 |

## 8) ゾーンを起動します。

ゾーンを起動し、動作確認をします。初期設定は必要ありません。

```
global02# zoneadm -z zone01 boot
```

## 9) ゾーンが正常に起動していることを確認します。

IP アドレスやホスト名などが引き継がれていること、およびサービス状態が正常であることなどを確認してください。

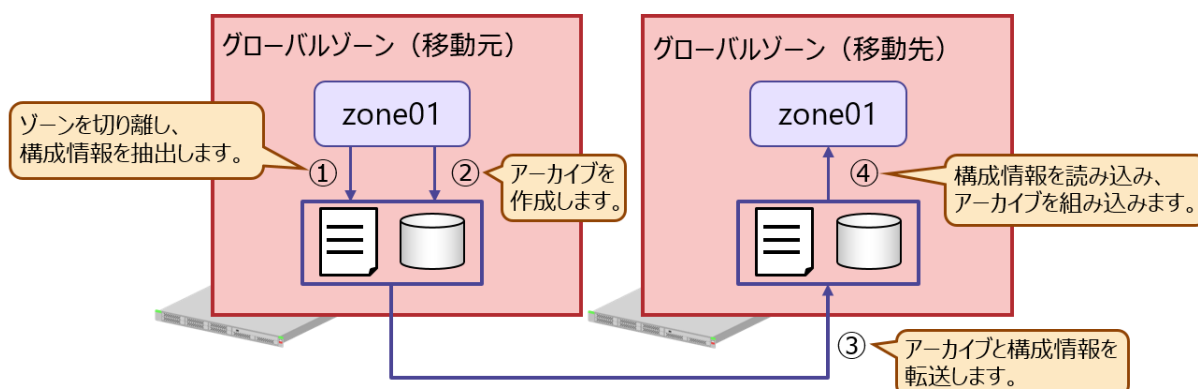
## 10) ゾーンを停止します。

```
global02# zoneadm -z zone01 shutdown
```

## 《注意》

本書の手順の場合、移動元環境と移動先環境の zone01 は IP アドレスが同一設定になるため、同時に起動させないでください。

## 5.2. ZFS アーカイブを使用したゾーンの移動



### 《注意》

ZFS アーカイブを使用したゾーンの移動は、Solaris 11.3 以前の環境が対象となります。

Solaris 11.4 の環境の場合は、「[5.1. 統合アーカイブを使用したゾーンの移動](#)」を実施してください。

### 5.2.1. ゾーンの切り離しと構成情報の抽出

#### 1) 移動元サーバ(global01)で、ゾーンを停止します。

- i) ゾーンを停止します。

```
global01# zoneadm -z zone01 shutdown
```

- ii) ゾーンの状態を表示します。

ゾーンが停止している (STATUS が「installed」になっている) ことを確認します。

```
global01# zoneadm list -vc
ID NAME          STATUS    PATH                                BRAND  IP
0  global         running   /                                  solaris shared
-  zone01         installed /zones/zone01                    solaris excl
```

#### 2) ゾーンを切り離し、移動できる状態にします。

```
global01# zoneadm -z zone01 detach
```

#### 3) ゾーンの状態を確認します。

ゾーンを切り離すと、ゾーンの STATUS が「installed」から「configured」に変わります。

```
global01# zoneadm list -vc
ID NAME          STATUS    PATH                                BRAND  IP
0  global         running   /                                  solaris shared
-  zone01         configured /zones/zone01                    solaris excl
```

#### 4) ディスク容量を確認します。

ゾーンのアーカイブ配置先(/zones)の容量を確認します。「AVAIL」(使用可能領域)が必要十分な値であることを確認してください。

```
global01# zfs list /zones
NAME      USED  AVAIL  REFER  MOUNTPOINT
zones     756M  547G   34.5K   /zones
```

#### 5) ゾーンの構成情報を複製します。

移動先で使用するゾーン構成情報を複製します。

```
global01# zonecfg -z zone01 export > /zones/zone01.zonecfg
```

### 5.2.2. アーカイブの作成と転送

#### 1) ゾーンのアーカイブデータを作成します。

ゾーンのデータセット(ファイルシステム)のスナップショットを作成し、スナップショットからアーカイブデータを作成します。

```
global01# zfs snapshot -r zones/zone01@v2v
global01# zfs send -rc zones/zone01@v2v | gzip > /zones/zone01.zfs.gz
```

#### 2) ゾーンを組み込んで、使用可能状態にします。

```
global01# zoneadm -z zone01 attach
Progress being logged to /var/log/zones/zoneadm.20190514T113645Z.zone01.attach
Installing: Using existing zone boot environment
Zone BE root dataset: zones/zone01/rpool/ROOT/solaris
Cache: Using /var/pkg/publisher.
Updating non-global zone: Linking to image /.
Processing linked: 1/1 done
Updating non-global zone: Auditing packages.
No updates necessary for this image. (zone:zone01)

Updating non-global zone: Zone updated.
Result: Attach Succeeded.
Log saved in non-global zone as
/zones/zone01/root/var/log/zones/zoneadm.20190514T113645Z.zone01.attach
```

#### Point

ゾーンの切り離し(detach)を実行すると、zonepath 上にゾーンの構成情報ファイル(SUNWAttached.xml)が作成されます。

ゾーンの構成情報ファイルには、ゾーンの構成情報やパッケージなどの情報が記載されており、ゾーンの組み込み(attach)時に利用されます。



### 3) 移動先(global02)のサーバへ ftp 接続し、アーカイブを転送します。

```
global01# ftp 192.168.10.ZZ
Connected to 192.168.10.ZZ
220 ::ffff:192.168.10.ZZ FTP server ready
Name (192.168.10.ZZ:user01): user01      ← 一般ユーザーでログイン
331 Password required for user01
Password: user01                        ← ログインユーザーのパスワードを入力（非表示）
230 User user01 logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> bin                                ← バイナリ転送モードを指定
200 Type set to I
ftp> cd /export/home/user01             ← 転送先ディレクトリの移動
250 CWD command successful
ftp> lcd /zones                          ← 転送元ディレクトリの移動
Local directory now /zones
ftp> !pwd                                ← 転送元ディレクトリの確認
/zones
ftp> put zone01.zonecfg                  ← zonecfg ファイルの転送
200 PORT command successful              (本書では/export/home/user01 配下に転送)
150 Opening BINARY mode data connection for zone01.zonecfg
226 Transfer complete
local: zone01.zonecfg remote: zone01.zonecfg
421 bytes sent in 0.0072 seconds (57.06 Kbytes/s)
ftp> put zone01.zfs.gz                  ← zone アーカイブの転送
200 PORT command successful
150 Opening BINARY mode data connection for zone01.zfs.gz
226 Transfer complete
local: zone01.zfs.gz remote: zone01.zfs.gz
280049098 bytes sent in 15 seconds (17944.95 Kbytes/s)
ftp> bye                                ← ftp の終了
221 Goodbye.
```

#### 《注意》

以降の手順で指定する zonepath と同じディレクトリにアーカイブファイルを置くと、ゾーンインストール時にエラーが発生します。本書の手順の場合は、/export/zones\_disaster 配下にアーカイブファイルを配置するとエラーが発生します。アーカイブファイルの転送先を変更する場合は、ご注意ください。

### 5.2.3. アーカイブの展開とゾーンの組み込み

#### 1) 移動元サーバ(global01)で、ゾーンの状態を確認します。

移動対象の zone01 が停止している (STATUS に installed が表示されている) ことを確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS    | PATH          | BRAND   | IP     |
|----|--------|-----------|---------------|---------|--------|
| 0  | global | running   | /             | solaris | shared |
| -  | zone01 | installed | /zones/zone01 | solaris | excl   |

#### 2) 移動先サーバ(global02)でアーカイブを展開し、組み込みます。

##### i) 複製した構成ファイルを編集します。

```
global02# vi /export/home/user01/zone01.zonecfg
```

編集後の/export/home/user01/zone01.zonecfg は、以下のとおりです。

移動元サーバ(global01)でリソースプール「pool\_1」を作成して zone01 に割り当てていた場合は、移動先サーバ(global02)で事前にリソースプール「pool\_1」を作成しておく必要があります。

※ 本書では事前に構成済みとしています。

```
create -b
set brand=solaris
set zonepath=/export/zones_disaster/zone01 ←/export/zones_disaster/zone01 に変更
set autoboot=false
set autoshutdown=shutdown
set pool=pool_1
set scheduling-class=FSS
set ip-type=exclusive
add anet
set linkname=vn1c0
set lower-link=net0
set configure-allowed-address=true
set link-protection=mac-nospoof
set mac-address=auto
end
```

##### ii) 新規ゾーンを構成します。

編集後の構成ファイルを元に、zone01 を再構成します。

```
global02# zonecfg -z zone01 -f /export/home/user01/zone01.zonecfg
```

##### iii) ゾーンの状態を確認します。

```
global02# zoneadm list -vc
```

| ID | NAME   | STATUS     | PATH                          | BRAND   | IP     |
|----|--------|------------|-------------------------------|---------|--------|
| 0  | global | running    | /                             | solaris | shared |
| -  | zone01 | configured | /export/zones_disaster/zone01 | solaris | excl   |

## iv) 移行したゾーンを組み込みます。

ゾーンの組み込みが成功すると「Attach Succeeded」と表示されます。

```
global02# zoneadm -z zone01 install -p -a /export/home/user01/zone01.zfs.gz
The following ZFS file system(s) have been created:
  rpool/export/zones_disaster
  rpool/export/zones_disaster/zone01
Progress being logged to /var/log/zones/zoneadm.20190514T120643Z.zone01.install
Installing: This may take several minutes...
  Zone BE root dataset: rpool/export/zones_disaster/zone01/rpool/ROOT/solaris-0
  Cache: Using /var/pkg/publisher.
Updating image format
Image format already current.
  Updating non-global zone: Linking to image /.
Processing linked: 1/1 done
  Updating non-global zone: Syncing packages.
No updates necessary for this image. (zone:zone01)

  Updating non-global zone: Zone updated.
  Result: Attach Succeeded.

  Done: Installation completed in 102.404 seconds.
  Next Steps: Boot the zone, then log into the zone console (zlogin -C)
               to complete the configuration process.
Log saved in non-global zone as
/export/zones_disaster/zone01/root/var/log/zones/zoneadm.20190514T120643Z.zone01.install
```

- ▶ ゾーンの組み込みには、数分かかります。
- ▶ 必須オプションとして「-p」または「-u」を指定する必要があります。  
「-p」: 元のゾーンのインストール情報(ホスト名、IP アドレスなど)を保持します。  
「-u」: 元のゾーンのインストール情報(ホスト名、IP アドレスなど)をクリア(sys-unconfig)します。ゾーン起動後、zlogin -C コマンドにて初期設定が必要です。
- ▶ ERROR メッセージが表示されますが、無視してください。ゾーンが起動されておらず、ZFS のマウントができないことを示すメッセージです。ZFS はゾーン起動時にマウントされるため、問題ありません。

## v) ゾーンの状態を確認します。

```
global02# zoneadm list -vc
```

| ID | NAME   | STATUS    | PATH                          | BRAND   | IP     |
|----|--------|-----------|-------------------------------|---------|--------|
| 0  | global | running   | /                             | solaris | shared |
| -  | zone01 | installed | /export/zones_disaster/zone01 | solaris | excl   |

## vi) リソースプールを確認します。

```
global02# poolstat -r all
```

| id | pool         | type | rid | rset         | min | max | size | used | load |
|----|--------------|------|-----|--------------|-----|-----|------|------|------|
| 1  | pool_1       | pset | 1   | pset_1       | 4   | 4   | 4    | 0.00 | 0.00 |
| 0  | pool_default | pset | -1  | pset_default | 4   | 66K | 4    | 0.00 | 0.59 |

vii) zone01 を起動します。

ゾーンを起動し、動作確認をします。初期設定は必要ありません。

```
global02# zoneadm -z zone01 boot
```

viii) 移動先のサーバで、zone01 のゾーンが正常に起動していることを確認します。

IP アドレスやホスト名などが引き継がれていること、およびサービス状態が正常であることなどを確認してください。

### 3) zone01 を停止します

```
global02# zoneadm -z zone01 shutdown
```

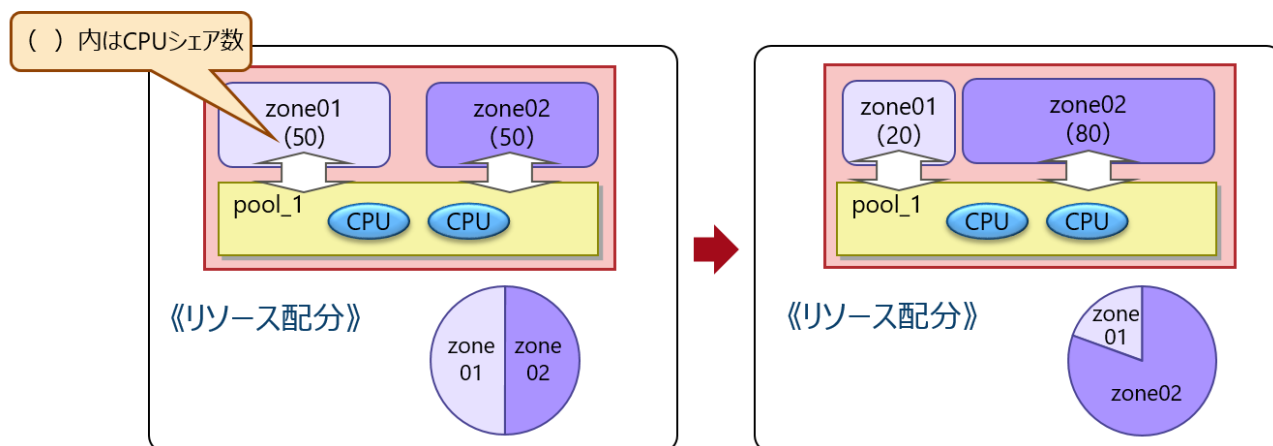
#### 《注意》

本書の手順の場合、移動元環境と移動先環境の zone01 は IP アドレスが同一設定になるため、同時に起動させないでください。

## 6. ゾーン(仮想 Solaris 環境)の CPU リソース操作

### 6.1. CPU シェア数の設定

各ゾーンが使用する CPU リソース配分比率を設定します。



#### 6.1.1. リソース操作の事前準備と確認

global01 にログインします。以降の手順はすべて global01 で実施します。

##### 1) ゾーンを起動します。

```
global01# zoneadm -z zone01 boot
```

##### 2) リソースの設定を確認します。

###### i) zone01 の CPU シェア数を確認します。

privileged 権限の VALUE の値を確認します。

```
global01# prctl -n zone.cpu-shares -i zone zone01
```

```
zone: 4: zone01
```

| NAME            | PRIVILEGE | VALUE | FLAG | ACTION | RECIPIENT |
|-----------------|-----------|-------|------|--------|-----------|
| zone.cpu-shares |           |       |      |        |           |
| usage           |           | 1     |      |        |           |
| privileged      |           | 1     | -    | none   | -         |
| system          |           | 65.5K | max  | none   | -         |

- ii) zone02 の CPU シェア数を確認します。

privileged 権限の VALUE の値を確認します。

```
global01# prctl -n zone.cpu-shares -i zone zone02
zone: 5: zone02
```

| NAME            | PRIVILEGE | VALUE | FLAG | ACTION | RECIPIENT |
|-----------------|-----------|-------|------|--------|-----------|
| zone.cpu-shares |           |       |      |        |           |
| usage           |           | 1     |      |        |           |
| privileged      |           | 1     | -    | none   | -         |
| system          |           | 65.5K | max  | none   | -         |

- iii) サービスを確認します。

svc:/system/pools:default サービスが「online」になっていることを確認します。

```
global01# svcs -a | grep pools
disabled Jan_07 svc:/system/pools/dynamic:default
online Jan_07 svc:/system/pools:default
```

- iv) リソースプールを確認します。

min、max、および size の値を確認します。

```
global01# poolstat -r all
```

| id | pool         | type | rid | rset         | min | max | size | used | load |
|----|--------------|------|-----|--------------|-----|-----|------|------|------|
| 1  | pool_1       | pset | 1   | pset_1       | 4   | 4   | 4    | 0.00 | 0.05 |
| 0  | pool_default | pset | -1  | pset_default | 4   | 66K | 44   | 0.00 | 0.02 |

▶ size 値は現在の CPU の数であり、必ず min 値と max 値の間に設定されます。

## 6.1.2. CPU シェア数の設定

### 1) CPU リソースの配分比率を設定します

本書では、zone01 に 20 %、zone02 に 80 % の比率で CPU リソースを配分します。

```
global01# prctl -n zone.cpu-shares -r -v 20 -i zone zone01
global01# prctl -n zone.cpu-shares -r -v 80 -i zone zone02
```

### 2) CPU シェア数を確認します。

#### ● zone01

```
global01# prctl -n zone.cpu-shares -i zone zone01
zone: 4: zone01
```

| NAME            | PRIVILEGE | VALUE | FLAG | ACTION | RECIPIENT |
|-----------------|-----------|-------|------|--------|-----------|
| zone.cpu-shares |           |       |      |        |           |
| usage           |           | 20    |      |        |           |
| privileged      |           | 20    | -    | none   | -         |
| system          |           | 65.5K | max  | none   | -         |

## ● zone02

```
global01# prctl -n zone.cpu-shares -i zone zone02
zone: 5: zone02
```

| NAME            | PRIVILEGE | VALUE | FLAG | ACTION | RECIPIENT |
|-----------------|-----------|-------|------|--------|-----------|
| zone.cpu-shares |           |       |      |        |           |
| usage           |           | 80    |      |        |           |
| privileged      |           | 80    | -    | none   | -         |
| system          |           | 65.5K | max  | none   | -         |

zone01 と zone02 の CPU シェア数が、「1:1」から「20:80」に変更されました。

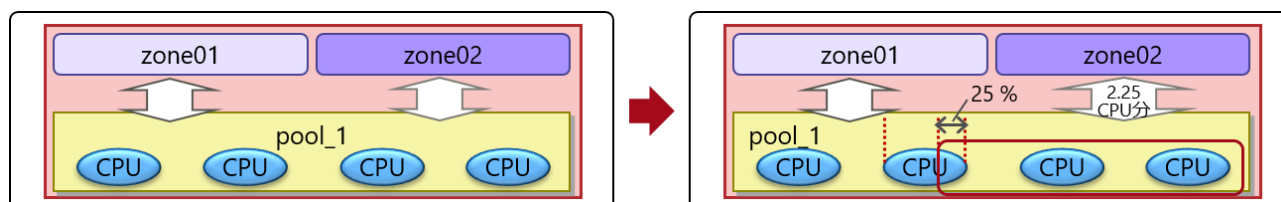
各ゾーンは、この CPU 配分比率に従って、共有しているリソースプールの CPU を利用します。

### Point

zone01 と zone02 に実際に負荷を発生させて prstat コマンドを使用すると、CPU リソースの配分状況を確認できます。

## 6.2. CPU キャップの設定

各ゾーンが使用する CPU リソースの上限値を設定します。



### 1) zone02 のリソース構成を確認します。

```
global01# zonecfg -z zone02 info capped-cpu
```

▶ キャッピングの設定がされていない場合、何も表示されません。

### 2) ゾーンのリソース構成情報の変更

#### i) CPU キャップを設定します。

zone02 のリソース設定に、CPU リソースの上限値を設定します。

ncpus パラメーターの値には、そのゾーンに占有させる CPU 数を小数第 2 位まで指定できます。

```
global01# zonecfg -z zone02
zonecfg:zone02> add capped-cpu
zonecfg:zone02:capped-cpu> set ncpus=2.25
zonecfg:zone02:capped-cpu> end
zonecfg:zone02> exit
```

▶ 設定の変更は、ゾーンの次回起動時に反映されます。

- ii) 設定を確認します。

```
global01# zonecfg -z zone02 info capped-cpu
capped-cpu:
    [ncpus: 2.25]
```

- iii) 別のパラメーターで設定を確認します。

```
global01# zonecfg -z zone02 info rctl
rctl:
    name: zone.cpu-cap
    value: (priv=privileged, limit=225, action=deny)
```

- ▶ capped-cpu パラメーターを指定すると、内部で zone.cpu-cap パラメーターが自動的に設定されます。
- ▶ zone.cpu-cap の limit パラメーターの値は、1 CPU 分のリソースを「100」として表示します。

- iv) zone02 の設定反映前の状態を確認します。

```
global01# prctl -n zone.cpu-cap -i zone zone02
zone: 5: zone02
```

| NAME         | PRIVILEGE | VALUE | FLAG | ACTION | RECIPIENT |
|--------------|-----------|-------|------|--------|-----------|
| zone.cpu-cap |           |       |      |        |           |
| usage        |           | 0     |      |        |           |
| system       |           | 4.29G | inf  | deny   | -         |

- v) zone02 を再起動し、設定を反映します。

```
global01# zoneadm -z zone02 reboot
```

- vi) zone02 の cpu-cap 数を確認します。

privileged 権限の VALUE の値が「225」になっていることを確認します。

```
global01# prctl -n zone.cpu-cap -i zone zone02
zone: 11: zone02
```

| NAME         | PRIVILEGE | VALUE | FLAG | ACTION | RECIPIENT |
|--------------|-----------|-------|------|--------|-----------|
| zone.cpu-cap |           |       |      |        |           |
| usage        |           | 18    |      |        |           |
| privileged   |           | 225   | -    | deny   | -         |
| system       |           | 4.29G | inf  | deny   | -         |

### 3) zone02 の cpu-cap 数を動的に変更します。

- i) cpu-cap 数を変更します。

```
global01# prctl -n zone.cpu-cap -r -v 375 -i zone zone02
```

- ▶ 動的変更を実施した場合、ゾーンの再起動後は元の値に戻ります。  
再起動後も変更を有効にする場合は、zonecfg コマンドでゾーンの定義情報を設定します。

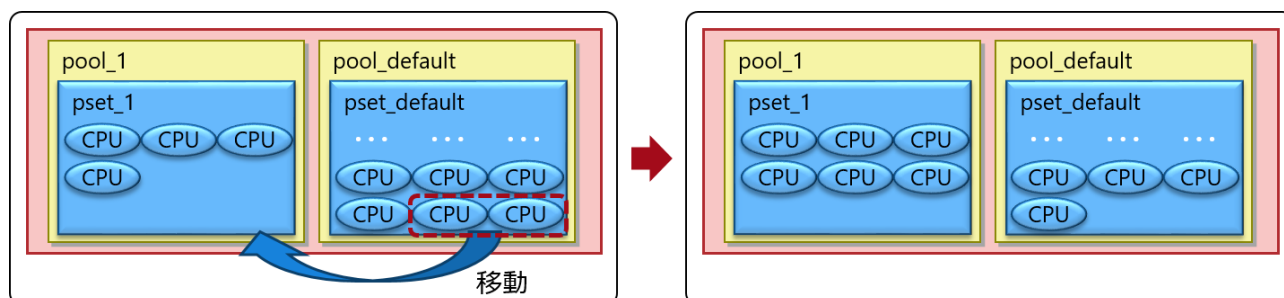


- ii) zone02 の cpu-cap 数を確認します。

```
global01# prctl -n zone.cpu-cap -i zone zone02
zone: 9: zone02
```

| NAME         | PRIVILEGE | VALUE | FLAG | ACTION | RECIPIENT |
|--------------|-----------|-------|------|--------|-----------|
| zone.cpu-cap |           |       |      |        |           |
| usage        |           | 0     |      |        |           |
| privileged   |           | 375   | -    | deny   | -         |
| system       |           | 4.29G | inf  | deny   | -         |

### 6.3. CPU リソースの動的移動



- 1) リソースプールを確認します。

size の値(実際に割り当たっている CPU 数)を確認します。

```
global01# poolstat -r all
```

| id | pool         | type | rid | rset         | min | max | size | used | load |
|----|--------------|------|-----|--------------|-----|-----|------|------|------|
| 1  | pool_1       | pset | 1   | pset_1       | 4   | 4   | 4    | 0.00 | 1.55 |
| 0  | pool_default | pset | -1  | pset_default | 4   | 66K | 44   | 0.00 | 0.24 |

- 2) pool\_default の最小 CPU スレッド数を 4 から 1 へ動的に変更します。

```
global01# poolcfg -dc 'modify pset pset_default (uint pset.min=1)'
```

- 3) pool\_1 の最大 CPU 数を 4 から 6 へ動的に変更します。

```
global01# poolcfg -dc 'modify pset pset_1 (uint pset.max=6)'
```

- 4) CPU リソースを移動します。

指定した数の CPU をプロセッサセット間で移動します。

```
global01# poolcfg -dc 'transfer 2 from pset pset_default to pset_1'
```

または、CPU ID を指定し、特定の CPU をプロセッサセット間で移動します。

```
global01# poolcfg -dc 'transfer to pset pset_1 (cpu 5; cpu 6)'
```

## 5) リソースプールの状態を確認します。

size の値（実際に割り当たっている CPU 数）を確認します。

```
global01# poolstat -r all
```

| id | pool         | type | rid | rset         | min | max | size | used | load |
|----|--------------|------|-----|--------------|-----|-----|------|------|------|
| 1  | pool_1       | pset | 1   | pset_1       | 4   | 6   | 6    | 0.00 | 0.13 |
| 0  | pool_default | pset | -1  | pset_default | 1   | 66K | 42   | 0.00 | 0.06 |

## 6) CPU リソースを移動します。

移動前の構成に戻します。

```
global01# poolcfg -dc 'transfer 2 from pset pset_1 to pset_default'
```

## 7) リソースプールの状態を確認します。

size の値（実際に割り当たっている CPU 数）を確認します。

```
global01# poolstat -r all
```

| id | pool         | type | rid | rset         | min | max | size | used | load |
|----|--------------|------|-----|--------------|-----|-----|------|------|------|
| 1  | pool_1       | pset | 1   | pset_1       | 4   | 6   | 4    | 0.00 | 0.06 |
| 0  | pool_default | pset | -1  | pset_default | 1   | 66K | 44   | 0.00 | 0.05 |

**《注意》**

上記の手順は、poolcfg コマンドに -d オプションを使用し、動的な変更のみ実施しています。そのため、リソースプールを再起動すると設定が元に戻ります。

継続した変更が必要な場合は、-d オプションを使用せずに poolcfg -c で変更を実施し、pooladm -c でリソースプール構成ファイル (pooladm.conf) を保存してください。

## 7. ゾーン(仮想 Solaris 環境)の仮想ネットワークの応用

### 7.1. 事前準備

#### 1) ネットワークサービスを確認します。

ネットワークサービスが起動している (STATE が「online」になっている) ことを確認します。

```
global01# svcs svc:/network/physical:default
STATE      STIME      FMRI
online      18:52:41   svc:/network/physical:default
```

#### 《参考》 Solaris 11 11/11 の場合

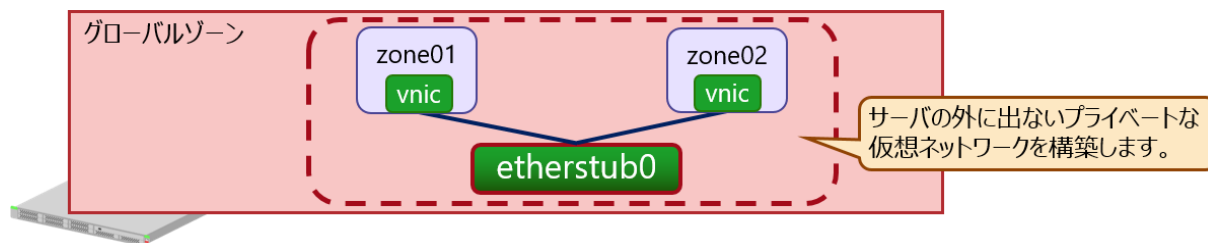
network auto-magic (nwam) サービスが停止している (STATE が「disabled」になっている) ことを確認します。

```
global01# svcs svc:/network/physical:nwam
STATE      STIME      FMRI
disabled    9月_20     svc:/network/physical:nwam
```

- ▶ nwam サービスは、システムの基本的なネットワーク設定を自動的に行う機能です。
- ▶ nwamd デーモンがプロファイルに従って自動的に設定を行ってしまうため、事前にサービスを停止します。

### 7.2. etherstub の設定

etherstub を作成し、2 つのゾーン (zone01 と zone02) のプライベート仮想ネットワークを構築します。



#### 7.2.1. etherstub の作成

##### 1) etherstub を作成します。

```
global01# dladm create-etherstub etherstub0
```

## 2) etherstub が作成されたことを確認します。

```
global01# dladm show-link
LINK          CLASS      MTU      STATE    OVER
etherstub0    etherstub 9000     unknown --
net0          phys      1500     up       --
net1          phys      1500     up       --
net2          phys      1500     unknown --
net3          phys      1500     unknown --
zone01/vnic0  vnic      1500     up       net0
zone02/vnic0  vnic      1500     up       net0
```

### 7.2.2. ゾーンに etherstub を割り当て

#### 1) 作成した etherstub を zone01 に割り当てます。

```
global01# zonecfg -z zone01
zonecfg:zone01> select anet lower-link=net0
zonecfg:zone01:anet> set lower-link=etherstub0
zonecfg:zone01:anet> end
zonecfg:zone01> exit
```

#### 2) 設定を確認します。

```
global01# zonecfg -z zone01 info -a anet
anet:
    linkname: vnic0
    lower-link: etherstub0
    allowed-address:
    configure-allowed-address: true
    defrouter:
    allowed-dhcp-cids:
    link-protection: mac-nospoof
    mac-address: auto
    auto-mac-address: 2:8:20:ea:23:8e
~（省略）~
```

#### 3) 作成した etherstub を zone02 に割り当てます。

```
global01# zonecfg -z zone02
zonecfg:zone02> select anet lower-link=net0
zonecfg:zone02:anet> set lower-link=etherstub0
zonecfg:zone02:anet> end
zonecfg:zone02> exit
```

- ▶ 設定の変更は、ゾーンの次回起動時に反映されます。

## 4) 設定を確認します。

```
global01# zonecfg -z zone02 info -a anet
anet:
    linkname: vnic0
    lower-link: etherstub0
    allowed-address:
    configure-allowed-address: true
    defrouter:
    allowed-dhcp-cids:
    link-protection: mac-nospoof
    mac-address: auto
    auto-mac-address: 2:8:20:fd:40:65
~（省略）~
```

## 5) インターフェースを確認します。

```
global01# dladm show-link
LINK          CLASS      MTU    STATE    OVER
etherstub0    etherstub 9000   unknown --
net0          phys      1500   up       --
net1          phys      1500   up       --
net2          phys      1500   unknown --
net3          phys      1500   unknown --
zone01/vnic0  vnic      1500   up       net0
zone02/vnic0  vnic      1500   up       net0
```

▶ この時点では、zone01、zone02 の VNIC は net0 に紐づいています。

## 6) グローバルゾーンからゾーンへ通信可能であることを確認します。

ここでは、zone01 と zone02 の IP アドレスを以下のとおりとしています。

- zone01 … 192.168.10.XX
- zone02 … 192.168.10.YY

```
global01# ping 192.168.10.XX
192.168.10.XX is alive
global01# ping 192.168.10.YY
192.168.10.YY is alive
```

## 7) ゾーンを再起動します。

```
global01# zoneadm -z zone01 reboot
global01# zoneadm -z zone02 reboot
```

## 8) ゾーン再起動後の状態を確認します。

ゾーンの STATUS が「running」になっていることを確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS  | PATH                      | BRAND   | IP     |
|----|--------|---------|---------------------------|---------|--------|
| 0  | global | running | /                         | solaris | shared |
| 3  | zone01 | running | /zones/zone01             | solaris | excl   |
| 4  | zone02 | running | /zones/zones_clone/zone02 | solaris | excl   |

## 9) 設定を確認します。

zone01 と zone02 の VNIC が、etherstub0 に紐づいていることを確認します。

```
global01# dladm show-link
```

| LINK         | CLASS     | MTU  | STATE   | OVER       |
|--------------|-----------|------|---------|------------|
| etherstub0   | etherstub | 9000 | unknown | --         |
| net0         | phys      | 1500 | up      | --         |
| net1         | phys      | 1500 | up      | --         |
| net2         | phys      | 1500 | unknown | --         |
| net3         | phys      | 1500 | unknown | --         |
| zone01/vnic0 | vnic      | 9000 | up      | etherstub0 |
| zone02/vnic0 | vnic      | 9000 | up      | etherstub0 |

## 10) グローバルゾーンからゾーンへ通信できないことを確認します。

```
global01# ping 192.168.10.XX
no answer from 192.168.10.XX
global01# ping 192.168.10.YY
no answer from 192.168.10.YY
```

## 7.2.3. ゾーン相互の疎通確認

## 1) ゾーンにログインします。

```
global01# zlogin zone01
```

▶ グローバルゾーンからの疎通がない状態でも、ログインは可能です。

## 2) zone01 から zone02 へ疎通確認をします。

```
zone01# ping 192.168.10.YY
192.168.10.YY is alive
```

▶ ゾーン相互には疎通が可能です。

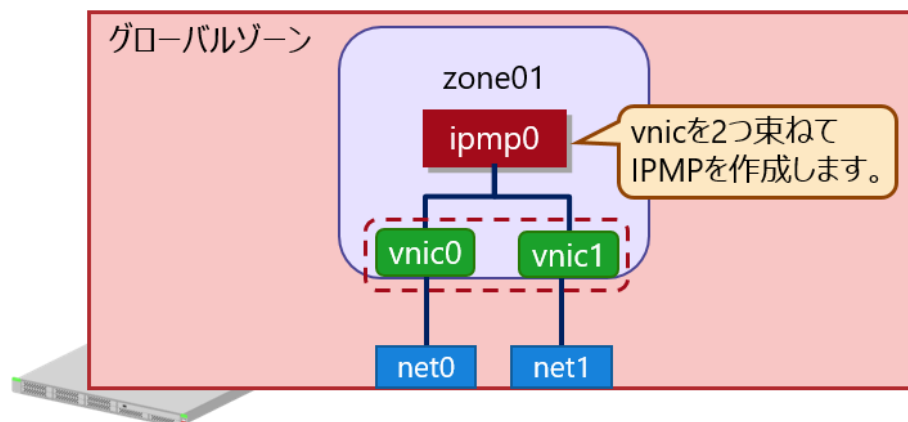
## 3) ゾーンからログアウトします。

```
zone01# exit
logout

[Connection to zone 'zone01' pts/1 closed]
global01#
```

### 7.3. IPMP の設定

IPMP を作成し、ゾーン(zone01)上で VNIC を冗長化します。



#### 7.3.1. ゾーン環境の準備

##### 1) lower-link を変更します。

lower-link パラメーターを「net0」に設定します。

```
global01# zonecfg -z zone01
zonecfg:zone01> select anet lower-link=etherstub0
zonecfg:zone01:anet> set lower-link=net0
zonecfg:zone01:anet> end
```

▶ 設定の変更は、ゾーンの次回起動時に反映されます。

##### 2) anet を追加します。

もう 1 つ VNIC を作成するため、anet を追加し、lower-link パラメーターに「net1」を設定します。

```
zonecfg:zone01> add anet
zonecfg:zone01:anet> set linkname=vnic1
zonecfg:zone01:anet> set lower-link=net1
zonecfg:zone01:anet> end
zonecfg:zone01> exit
```

▶ 設定の変更は、ゾーンの次回起動時に反映されます。

### 3) anet が「net0」と「net1」に設定されていることを確認します。

```
global01# zonecfg -z zone01 info -a anet
anet:
    linkname: vnic0
    lower-link: net0
    allowed-address:
    configure-allowed-address: true
~ (省略) ~
anet:
    linkname: vnic1
    lower-link: net1
    allowed-address:
    configure-allowed-address: true
~ (省略) ~
```

### 4) ゾーンを再起動します。

```
global01# zoneadm -z zone01 reboot
```

### 5) ゾーン再起動後の状態を確認します。

ゾーンの STATUS が「running」になっていることを確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS  | PATH                      | BRAND   | IP     |
|----|--------|---------|---------------------------|---------|--------|
| 0  | global | running | /                         | solaris | shared |
| 4  | zone02 | running | /zones/zones_clone/zone02 | solaris | excl   |
| 5  | zone01 | running | /zones/zone01             | solaris | excl   |

### 6) 設定を確認します。

zone01 の 2 つの VNIC が、「net0」と「net1」に紐づいていることを確認します。

```
global01# dladm show-link
```

| Link Name    | Link Type | MTU  | State   | Lower Link |
|--------------|-----------|------|---------|------------|
| net0         | phys      | 1500 | up      | ---        |
| net1         | phys      | 1500 | up      | ---        |
| net2         | phys      | 1500 | unknown | ---        |
| net3         | phys      | 1500 | unknown | ---        |
| ~ (省略) ~     |           |      |         |            |
| zone01/vnic0 | vnic      | 1500 | up      | net0       |
| zone01/vnic1 | vnic      | 1500 | up      | net1       |



### 7.3.2. IPMP の作成

- 1) ゾーンにログインします。

```
global01# zlogin zone01
```

- 2) IPMP インターフェースを作成します。

```
zone01# ipadm create-ipmp ipmp0
```

- 3) ipmp0 が作成されていることを確認します。

```
zone01# ipadm
NAME                CLASS/TYPE STATE    UNDER  ADDR
ipmp0              ipmp      failed   --      --
~ (省略) ~
vnic0               ip        ok       --      --
  vnic0/v4          static    ok       --      192.168.10.XX/24
  vnic0/v6          addrconf ok       --      fe80::8:20ff:feea:238e/10
```

- 4) IPMP に組み込むために、vnic インターフェースのアドレスを削除します。

```
zone01# ipadm delete-addr vnic0/v4
zone01# ipadm delete-addr vnic0/v6
```

- 5) IPMP に組み込む vnic インターフェースを追加します。

```
zone01# ipadm create-ip vnic1
```

- 6) vnic0 と vnic1 のインターフェースがあり、それぞれにアドレスが設定されていないことを確認します。

```
zone01# ipadm
NAME                CLASS/TYPE STATE    UNDER  ADDR
ipmp0              ipmp      failed   --      --
~ (省略) ~
vnic0            ip        down     --      --
vnic1            ip        down     --      --
```

- 7) IPMP へインターフェースを組み込みます。

```
zone01# ipadm add-ipmp -i vnic0 -i vnic1 ipmp0
```

- 8) 「vnic0」と「vnic1」が ipmp0 に組み込まれたことを確認します。

```
zone01# ipadm
NAME                CLASS/TYPE STATE    UNDER  ADDR
ipmp0              ipmp      down     --      --
~ (省略) ~
vnic0             ip        ok       --      ipmp0
vnic1             ip        ok       --      ipmp0
```

## 9) IP アドレスを設定します。

vnic0 に設定していた IP アドレスを ipmp0 に設定します。

```
zone01# ipadm create-addr -T static -a local=192.168.10.XX/24 ipmp0/v4
```

## 10) ipmp0 インターフェースに IP アドレスが設定されたことを確認します。

```
zone01# ipadm
NAME          CLASS/TYPE STATE    UNDER  ADDR
ipmp0         ipmp      ok       --      --
  ipmp0/v4    static   ok       --      192.168.10.XX/24
~ (省略) ~
vnic0         ip        ok       ipmp0   --
vnic1         ip        ok       ipmp0   --
```

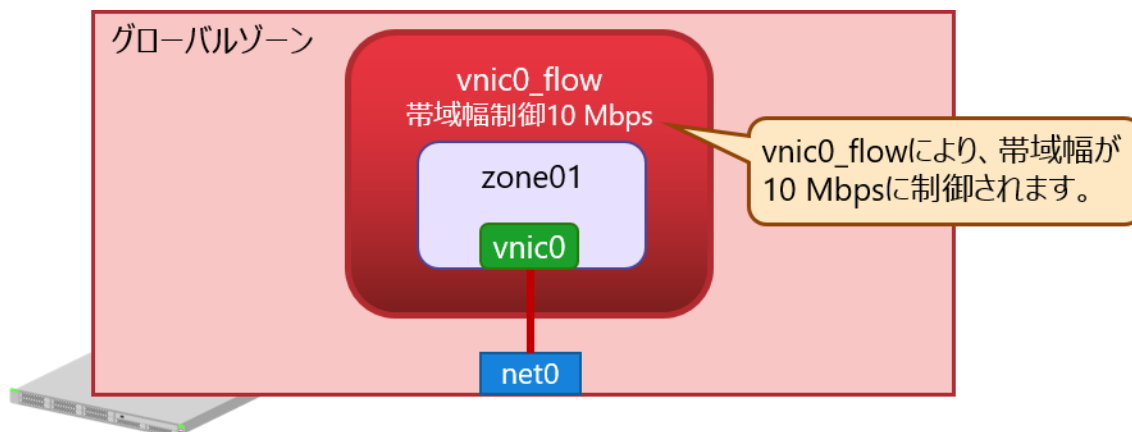
本手順を実施した場合、「分散型構成」で「リンクベース」の IPMP が作成されます。

IPMP のその他の構成や詳細については、以下の Oracle 社のマニュアルを参照してください。

[https://docs.oracle.com/cd/E37838\\_01/html/E60991/gfkey.html#scrolltoc](https://docs.oracle.com/cd/E37838_01/html/E60991/gfkey.html#scrolltoc)

## 7.4. 帯域幅制限の設定

フローを作成し、ゾーン(zone01)の仮想 NIC(vnic0)の帯域幅を 10 Mbps に制限します。



### 7.4.1. ゾーン環境の準備

- 1) 「[7.3. IPMP の設定](#)」で作成した IPMP を削除します。

```
zone01# ipadm delete-addr ipmp0/v4
zone01# ipadm remove-ipmp -i vnic0 -i vnic1 ipmp0
zone01# ipadm delete-ipmp ipmp0
zone01# ipadm delete-ip vnic1
```

- 2) vnic0 に IP アドレスを設定します。

```
zone01# ipadm create-addr -T static -a local=192.168.10.XX/24 vnic0/v4
```

### 7.4.2. フローの作成

- 1) 制限を設ける仮想 NIC ごとに、フローを作成します。

【書式】flowadm add-flow -l <リンク名> -a local\_ip=<IP アドレス> <フロー名>

```
zone01# flowadm add-flow -l vnic0 -a local_ip=192.168.10.XX vnic0_flow
```

▶ IP アドレスは、フローを作成する VNIC の IP アドレスを指定します。ここでは、vnic0 の IP アドレスを指定します。

- 2) vnic0\_flow が作成されていることを確認します。

```
zone01# flowadm show-flow
```

| FLOW       | LINK  | PROTO | LADDR         | LPORT | RADDR | RPORT | DIR |
|------------|-------|-------|---------------|-------|-------|-------|-----|
| vnic0_flow | vnic0 | --    | 192.168.10.XX | --    | --    | --    | bi  |

### 3) フローのプロパティ(帯域幅制限)を設定します。

VNIC(vnic0)に対して 10 Mbps(1.25 MB/s)の制限を実施します。

```
zone01# flowadm set-flowprop -p max-bw=10M vnic0_flow
```

- ▶ max-bw 値は、Kbps、Mbps、または Gbps を示す K、M、または G を付けて指定できます。指定しない場合、Mbps となります。

#### 《注意》

Oracle Solaris 11.3 以前の場合は、max-bw プロパティを maxbw プロパティに置き換えてください。

```
zone01# flowadm set-flowprop -p maxbw=10M vnic0_flow
```

### 4) 設定した帯域幅制限が VALUE に表示されることを確認します。

```
zone01# flowadm show-flowprop -p max-bw
```

| FLOW       | PROPERTY | PERM | VALUE | EFFECTIVE | DEFAULT | POSSIBLE |
|------------|----------|------|-------|-----------|---------|----------|
| vnic0_flow | max-bw   | rw   | 10    | 10        | --      | --       |

- ▶ VALUE の値の単位は Mbps です。最低値は 10 Kbps になります。

#### 《参考》フロープロパティの設定を初期化／削除するには

- フロープロパティの設定を初期化

【書式】flowadm reset-flowprop [-t] [-R ルートディレクトリ] [-p プロパティ名] <フロー名>

```
zone01# flowadm reset-flowprop vnic0_flow
```

- フロープロパティの設定を削除

【書式】flowadm remove-flow [-t] [-R ルートディレクトリ] <フロー名>

```
zone01# flowadm remove-flow vnic0_flow
```

### 《参考》帯域幅制限の確認

ほかのホストから FTP などを使用し、ISO ファイルなどサイズが大きいファイルをゾーンへ転送することで確認できます。

ここでは例として、グローバルゾーンから zone01 へ ISO ファイルを転送している最中に、flowstat コマンドで帯域幅制限の確認をします。

※ zone01 に FTP がインストールされていない場合は、FTP のインストールとサービスの有効化を実施してください。例を以下に示します。

```
zone01# pkg install pkg://solaris/service/network/ftp
zone01# svcadm enable svc:/network/ftp:default
```

(1) グローバルゾーンから zone01 へ ftp 接続し、ISO ファイルの転送を開始します。

(2) zone01 上で以下のコマンドを実行し、帯域幅制限の動作を確認します。

【書式】flowstat <フロー名> [interval(秒) [回数] ]

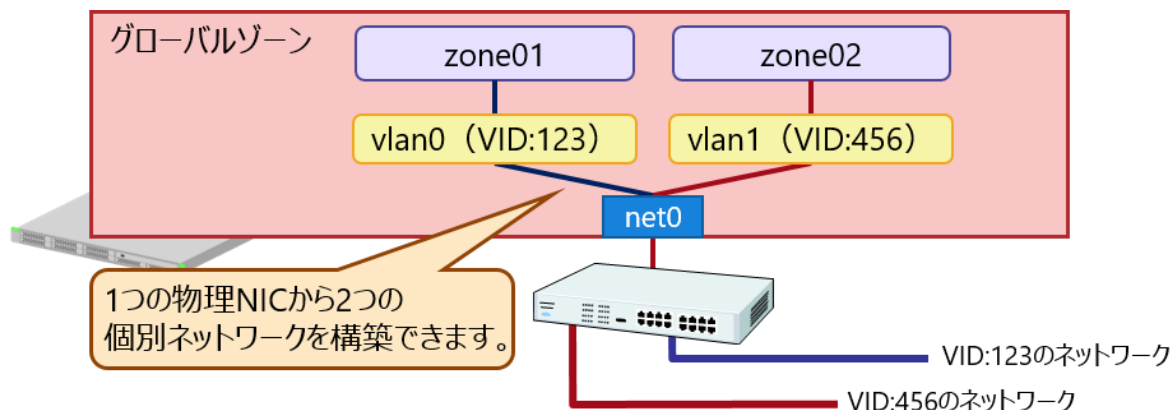
```
zone01# flowstat vnic0_flow 1
      FLOW      IPKTS    RBYTES    IDROPS    OPKTS    OBYTES    ODROPS
vnic0_flow    1.00K    1.35M        68      517    34.57K         0
vnic0_flow      907    1.21M        77      478    32.22K         0
vnic0_flow     987    1.37M        80      521    35.30K         0
vnic0_flow     785    1.10M        62      409    27.35K         0
:
~ (省略) ~
```

- ▶ 回数を指定しない場合は、無期限に出力します。[Ctrl]+[C]キーを押して停止します。
- ▶ RBYTES が転送量 (MB/s) を示します。
- ▶ flowadm コマンドで帯域幅を 10 Mbps に設定した場合、flowstat コマンドの出力結果は約 1.25 MB/s の転送量となります (ビットからバイト単位に変換)。

## 7.5. VLAN の設定

VLAN を作成し、1 つの物理 NIC (net0) から 2 つの異なるネットワークを構築します。

本書では、exclusive 方式の net リソースを使用する設定例を記載しています。



### 7.5.1. ゾーン環境の準備

- 1) 「[7.4. 帯域幅制限の設定](#)」で作成したフローを削除し、グローバルゾーンへ戻ります。

```
zone01# flowadm remove-flow vnic0_flow
zone01# exit
logout

[Connection to zone 'zone01' pts/1 closed]
global01#
```

- 2) ゾーンの anet リソースを削除します。

- zone01

```
global01# zonecfg -z zone01
zonecfg:zone01> remove anet 0
zonecfg:zone01> remove anet 1
zonecfg:zone01> exit
```

- zone02

```
global01# zonecfg -z zone02
zonecfg:zone02> remove anet 0
zonecfg:zone02> exit
```

## 7.5.2. VLAN の作成

### 1) VLAN インターフェース作成

ここでは、以下の VLAN を作成します。

| 項番 | VLAN 名 | VID | 割り当てるゾーン | 紐づける物理 NIC |
|----|--------|-----|----------|------------|
| 1  | vlan0  | 123 | zone01   | net0       |
| 2  | vlan1  | 456 | zone02   | net0       |

```
global01# dladm create-vlan -l net0 -v 123 vlan0
global01# dladm create-vlan -l net0 -v 456 vlan1
```

### 2) net0 に紐づいた vlan0 と vlan1 が作成されたことを確認します。

```
global01# dladm show-link
LINK          CLASS      MTU    STATE    OVER
net0          phys      1500   up       --
net1          phys      1500   up       --
net2          phys      1500   unknown --
net3          phys      1500   unknown --
vlan0         vlan      1500   up       net0
vlan1         vlan      1500   up       net0
~ (省略) ~
```

### 3) VLAN をゾーンに割り当てます。

ここでは、zone01 に vlan0、zone02 に vlan1 を割り当てます。

- zone01

```
global01# zonecfg -z zone01
zonecfg:zone01> add net
zonecfg:zone01:net> set physical=vlan0
zonecfg:zone01:net> end
zonecfg:zone01> exit
```

- zone02

```
global01# zonecfg -z zone02
zonecfg:zone02> add net
zonecfg:zone02:net> set physical=vlan1
zonecfg:zone02:net> end
zonecfg:zone02> exit
```

- ▶ 設定の変更は、ゾーンの次回起動時に反映されます。

## 4) 設定を確認します。

- zone01

```
global01# zonecfg -z zone01 info -a net
net:
    address:
    allowed-address:
    configure-allowed-address: true
    physical: vlan0
    defrouter:
```

- zone02

```
global01# zonecfg -z zone02 info -a net
net:
    address:
    allowed-address:
    configure-allowed-address: true
    physical: vlan1
    defrouter:
```

## 5) ゾーンを再起動します。

```
global01# zoneadm -z zone01 reboot
global01# zoneadm -z zone02 reboot
```

## 6) ゾーン再起動後の状態を確認します。

ゾーンの STATUS が「running」になっていることを確認します。

```
global01# zoneadm list -vc
```

| ID | NAME   | STATUS         | PATH                      | BRAND   | IP     |
|----|--------|----------------|---------------------------|---------|--------|
| 0  | global | running        | /                         | solaris | shared |
| 7  | zone01 | <b>running</b> | /zones/zone01             | solaris | excl   |
| 8  | zone02 | <b>running</b> | /zones/zones_clone/zone02 | solaris | excl   |

## 7) VLAN の設定を確認します。

作成した vlan がそれぞれのゾーンに割り当たっていることを確認します。

```
global01# dladm show-vlan -z zone01,zone02
```

| LINK                | VID        | SVID | PVLAN-TYPE | FLAGS | OVER        |
|---------------------|------------|------|------------|-------|-------------|
| <b>zone01/vlan0</b> | <b>123</b> | --   | --         | ----  | <b>net0</b> |
| <b>zone02/vlan1</b> | <b>456</b> | --   | --         | ----  | <b>net0</b> |

以上で、サーバ側の VLAN 設定は完了です。

このあと、作成した VLAN の VID をスイッチ側に設定する必要があります。



## 《参考》VLAN の削除手順

(1) VLAN の設定を確認します。

vlan がそれぞれのゾーンに割り当たっていることを確認します。

```
global01# dladm show-vlan
LINK          VID  SVID PVLAN-TYPE  FLAGS  OVER
vlan0         123  --   --          ----- net0
zone01/vlan0  123  --   --          ----- net0
vlan1         456  --   --          ----- net0
zone02/vlan1  456  --   --          ----- net0
```

(2) ゾーンの net リソースを削除します。

- zone01

```
global01# zonecfg -z zone01
zonecfg:zone01> remove net 0
zonecfg:zone01> exit
```

- zone02

```
global01# zonecfg -z zone02
zonecfg:zone02> remove net 0
zonecfg:zone02> exit
```

(3) ゾーンを再起動します。

```
global01# zoneadm -z zone01 reboot
global01# zoneadm -z zone02 reboot
```

(4) ゾーン再起動後の状態を確認します。

ゾーンの STATUS が「running」になっていることを確認します。

```
global01# zoneadm list -vc
ID NAME          STATUS    PATH                                BRAND  IP
0  global         running   /                                  solaris shared
9  zone01         running   /zones/zone01                     solaris excl
10 zone02         running   /zones/zones_clone/zone02         solaris excl
```

(5) VLAN の設定を確認します。

vlan がゾーンに割り当たっていないことを確認します。

```
global01# dladm show-vlan
LINK          VID  SVID PVLAN-TYPE  FLAGS  OVER
vlan0         123  --   --          ----- net0
vlan1         456  --   --          ----- net0
```

(6) VLAN を確認します。

```
global01# dladm show-link
LINK          CLASS    MTU    STATE    OVER
net0          phys     1500   up       --
net1          phys     1500   up       --
net2          phys     1500   unknown  --
net3          phys     1500   unknown  --
vlan0         vlan     1500   up       net0
vlan1         vlan     1500   up       net0
~ (省略) ~
```

(7) VLAN を削除します。

```
global01# dladm delete-vlan vlan0
global01# dladm delete-vlan vlan1
```

(8) VLAN が削除されたことを確認します。

```
global01# dladm show-link
LINK          CLASS    MTU    STATE    OVER
net0          phys     1500   up       --
net1          phys     1500   up       --
net2          phys     1500   unknown  --
net3          phys     1500   unknown  --
~ (省略) ~
```

## 改版履歴

| 改版日         | 版数    | 改版内容                        |
|-------------|-------|-----------------------------|
| 2013 年 7 月  | 2.0 版 | Oracle Solaris 11 11/11 に対応 |
| 2014 年 5 月  | 2.1 版 | Oracle Solaris 11.1 に対応     |
| 2016 年 10 月 | 2.2 版 | Oracle Solaris 11.3 に対応     |
| 2017 年 4 月  | 2.3 版 | レイアウトデザインと構成を更新             |
| 2019 年 6 月  | 3.0 版 | Oracle Solaris 11.4 に対応     |

