

NEC Scalable Technology File System for AI (ScaTeFS for AI) インストールガイド

輸出する際の注意事項

本製品（ソフトウェアを含む）は、外国為替および外国貿易法で規定される規制貨物（または役務）に該当することがあります。

その場合、日本国外へ輸出する場合には日本国政府の輸出許可が必要です。

なお、輸出許可申請手続きにあたり資料等が必要な場合には、お買い上げの販売店またはお近くの当社営業拠点にご相談ください。

は し が き

このマニュアルは、NEC Scalable Technology File System for AI 用のインストールガイドです。ここではマニュアルの目的、対象読者について説明します。

マニュアルの目的

このマニュアルでは、NEC Scalable Technology File System for AI を利用するために必要な IO サーバ、クライアントのインストールの詳細情報や具体的な設定手順等に関する説明を目的としています。

対象読者

このマニュアルは、次の方を対象読者として記述しています。

- システム管理者

備考

- (1) Linux は Linus Torvalds氏 の日本およびその他の国における登録商標あるいは商標です。
- (2) Red Hatは米国およびその他の国でのRed Hat, Inc. の登録商標もしくは商標です。
- (3) CLUSTERPROは日本電気株式会社の登録商標です。
- (4) Windowsは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。
- (5) NVIDIAは、米国およびその他の国におけるNVIDIA Corporationの登録商標または商標です。
- (6) その他記載の会社名、製品名は、それぞれの会社の商標、もしくは登録商標です。

本書の読み進め方

本書は、次の構成となっています。章ごとに対象読者の範囲は、表の一番右の列にその範囲を示しています。

章	タイトル	内容	対象読者
1	はじめに	ソフトウェアの入手方法やシステム要件について記載しています。	システム管理者
2	IOサーバのセットアップ	IOサーバの設計、ディストリビューションおよび各種ソフトウェアのインストール、ScaTeFSのセットアップや初期設定等について記載しています。	システム管理者
3	Linuxクライアントのセットアップと運用	Linuxクライアントのセットアップと運用方法、アップデート方法について記載しています。	システム管理者
4	ストレージ (V100) のセットアップと運用	IOサーバがアクセスするストレージ (iStorage V100) のセットアップ方法と運用方法について記載しています。	システム管理者

関連説明書

- 『NEC Scalable Technology File System for AI (ScaTeFS for AI) システムガイド』
- 『NEC Scalable Technology File System for AI (ScaTeFS for AI) ユーザーズガイド』
- 『NEC Scalable Technology File System for AI (ScaTeFS for AI) ソフトウェアライセンス管理説明書』

各種説明書は以下のNECサポートポータル/Webページから参照できます。

<https://www.support.nec.co.jp/View.aspx?id=3170102881>

用語定義・略語

用語	意味
IOサーバ	ScaTeFSを構成するサーバ。最低 2 台必要。
LV	論理ボリュームの略称。
LVM	「Logical Volume Manager」の略称。
NFS	「Network File System」の略称。
PV	物理ボリュームの略称。
RHEL	「Red Hat Enterprise Linux」の略称。
ScaTeFS for AI	「NEC Scalable Technology File System for AI」の略称。
VG	ボリュームグループの略称。
プリマップ	指定されたファイルサイズに相当する個数の実ファイルを各実ファイルシステム上に予め生成する機能のこと。同ファイルへの並列 writeを行う場合、実ファイル生成のオーバーヘッドをプリマップにより低減することが目的。scatefs_premapを使用。
ルートIOサーバ	IOサーバの一種。mkfsを実行するサーバであり、クライアントがマウントする際のマウント先のサーバ。システム運用中においては、他のIOサーバと特に相違はなく同様な処理を行う。
仮想ファイル	仮想ファイルシステム上に作成されたファイル。ScaTeFS上のレギュラーファイル。
仮想ファイルシステム	複数のIOターゲットにより構成されるクライアント見えのファイルシステム。ScaTeFSそのもの。
実ファイル	複数のサーバに跨った仮想ファイルの断片。実際には、実ファイルシステム上のファイルのこと。
実ファイルシステム、IOターゲット	仮想ファイルシステムを構成する基本単位。各IOサーバ配下に作成される。実態は、Linuxで使用可能な通常のファイルシステム。

目 次

第 1 章	はじめに	1
1.1	ソフトウェアの入手方法	1
1.2	システム要件	1
第 2 章	IO サーバのセットアップ	2
2.1	設計	2
2.1.1	ストレージのディスク構成	2
2.1.2	LVM と IO ターゲット	2
2.1.3	ネットワーク	4
2.1.4	CLUSTERPRO クラスタ構成情報の設計と作成	4
2.2	ディストリビューションおよび各種ソフトウェアのインストール	4
2.2.1	Linux ディストリビューションのインストール	4
2.2.2	RHEL のリポジトリの設定	5
2.2.3	時刻同期	6
2.2.4	HA Dynamic Link Manager	6
2.2.5	CLUSTERPRO	6
2.2.6	リモートシェル(rsh)	6
2.2.7	ScaTeFS/Server	7
2.2.8	ルーティング	8
2.2.9	Zabbix/Agent	8
2.3	セットアップ	9
2.3.1	GRUB2 の設定	9
2.3.2	ファイアウォール	10
2.3.3	ネットワーク	11
2.3.3.1	ファイルシステムポート	11
2.3.3.2	IO サーバ間インタコネク用ポート	12
2.3.3.3	ファイルシステムポートのルーティング設定	13
2.3.4	ファイルシステム管理アカウント	14
2.3.5	内蔵ディスク (ScaTeFS 用) の設定	14
2.3.6	カーネルパラメータ	14
2.3.7	systemd の設定	15
2.3.8	sysstat の設定	15
2.3.9	crashdump の設定	16

2.4	ScaTeFS のセットアップ	17
2.4.1	IO サーバの登録	17
2.4.2	HDLM デバイス名の確認	18
2.4.3	パーティションの作成	18
2.4.4	LVM デバイスの作成	19
2.4.5	IO ターゲットの登録	22
2.4.6	ScaTeFS のファイルシステム作成	23
2.4.7	ScaTeFS ライセンス	25
2.4.8	CLUSTERPRO	25
2.5	アップデート	26
2.5.1	システム運用を停止して行うアップデート	26
2.5.2	システム運用を継続して行うアップデート (無停止アップデート)	27
第3章	Linux クライアントのセットアップと運用	29
3.1	クライアントのインストール	29
3.1.1	Linux ディストリビューションのインストール	29
3.1.2	カーネルクラッシュダンプ採取の有効化	29
3.1.3	ScaTeFS クライアントのパッケージを含む zip ファイルの入手	29
3.1.4	ScaTeFS クライアントのパッケージのインストール	30
3.1.5	インストール対象マシンの再起動	30
3.2	クライアントのセットアップ	30
3.2.1	ルーティング設定	30
3.2.2	マウント方法	31
3.2.3	全 IO サーバとのコネクション確立の確認	32
3.2.4	アンマウント方法	33
3.3	クライアントのアップデート	33
3.3.1	ファイルシステムのアンマウントと scatefs-client サービスの停止	33
3.3.2	ScaTeFS クライアントのパッケージのアンインストール	34
3.3.3	カーネルのアップデート (選択)	34
3.3.4	ScaTeFS クライアントのパッケージを含む zip ファイルの入手	34
3.3.5	ScaTeFS クライアントのパッケージのインストール	35
3.3.6	アップデート対象マシンの再起動	35
3.3.7	ScaTeFS クライアントのセットアップ	35
3.4	クライアントのアンインストール	35
3.4.1	ファイルシステムのアンマウントと scatefs-client サービスの停止	35

3.4.2 ScaTeFS クライアントのパッケージのアンインストール	36
第 4 章 ストレージ (V100) のセットアップと運用.....	37
4.1 ストレージ (V100) のセットアップ	37
4.1.1 IO ターゲットの設計	37
4.1.2 ストレージの設定	37
4.1.2.1 パリティグループの作成.....	38
4.1.2.2 プール LDEV の作成	38
4.1.2.3 プールの作成.....	39
4.1.2.4 仮想ボリュームの作成	40
4.1.2.5 ホストグループ作成	41
4.1.2.6 LUN パスの追加	42
4.1.2.7 スペアドライブの割り当て.....	42
4.1.3 LVM の設定.....	42
付録 A CLUSTERPRO のクラスタ構成情報作成手順 (オフラインバージョン).....	44
A.1 はじめに	44
A.2 CLUSTERPRO ツールのインストール.....	45
A.3 CLUSTERPRO ツールの起動	45
A.4 クラスタ構成情報作成	45
A.5 クラスタ	45
A.6 サーバの基本設定.....	45
A.7 サーバのインタコネクタの設定.....	46
A.8 サーバのフェンシングの設定.....	46
A.9 フェイルオーバーグループの作成.....	46
A.10 フェイルオーバーグループの追加.....	47
A.11 グループリソース (フローティング IP リソース) の追加	47
A.12 グループリソース (ボリュームマネージャリソース) の追加	48
A.13 グループリソース (ディスクリソース) の追加	49
A.14 グループリソース (EXEC リソース) の追加	51
A.15 モニタリソース (ディスクモニタ) 追加.....	53
A.16 モニタリソース (カスタムモニタ) 追加.....	54
A.17 モニタリソース (ボリュームマネージャモニタ) 設定変更	55
A.18 モニタリソース (ユーザ空間モニタ) 設定変更	55
A.19 モニタリソース (フローティング IP モニタ) 設定変更.....	56
A.20 モニタリソース (IP モニタ)の追加	56

A.21	モニタリソース異常時の回復動作設定	58
A.22	クラスタプロパティの変更	58
A.23	クラスタ構成情報のパラメーター一覧	58
付録 B	発行履歴	66
B.1	発行履歴一覧表.....	66
B.2	追加・変更点詳細.....	66

表目次

表 2-1	1 台のストレージのディスク構成	2
表 2-2	ScaTeFS データ領域のストレージ仮想ボリュームと LVM (VG) 構成.....	3
表 2-3	ScaTeFS メタデータ領域のストレージ仮想ボリュームと LVM (VG) 構成 ..	3
表 2-4	scatefs_addios の設定項目	17
表 2-5	scatefs_addiot の設定項目.....	22
表 2-6	scatefs_mkfs の設定項目	24
表 2-7	IO ターゲットを使用する順序	24
表 2-8	iotid の設定値	24
表 3-1	使用できる Linux ディストリビューションとカーネルバージョン	29

第1章 はじめに

本書では、NEC Scalable Technology File System (ScaTeFS) のインストール及び設定について説明します。

1.1 ソフトウェアの入手方法

「インターネット配信 製品ダウンロードサービス」より入手してください。

(<https://idpsv.wsrs.m-dmz1.nec.co.jp/dlservice/>)

修正物件が出ている場合は「NECサポートポータル」より入手できます。

1.2 システム要件

システム要件は以下の通りです。サポート対象のRHELバージョン とカーネルバージョンについては NEC サポートポータルをご確認ください。

	機種等	ディストリビューション
Linux クライアント	x86-64 アーキテクチャをサポートする Linux マシン	RHEL
IO サーバ	Express5800/R120j-2M	RHEL

第2章 IO サーバのセットアップ

2.1 設計

IOサーバを構築する前にLVMとIOターゲット、ネットワーク、CLUSTERPROクラスタ構成の設計を実施します。

2.1.1 ストレージのディスク構成

ストレージに搭載するディスクについて、ディスク構成の代表例を記載します。本マニュアルでは、下記ディスク構成のストレージを2台使用する例で説明を記載します。

表 2-1 1 台のストレージのディスク構成

用途	NVMe SSD		パリティグループ			仮想 ボリューム
	容量	ドライブ数	RAID	数	LDEV	
データ 領域	1.92TB	16	RAID6 (14D+2P)	1	4	4
メタデータ 領域		4	RAID1 (2D+2D)	1	4	4

2.1.2 LVM と IO ターゲット

IOターゲットは、ScaTeFSのファイルシステムの基盤となるデータストアです。クライアントノードから書き込まれたファイルデータは、IOサーバに分散され、さらにIOサーバ内のIOターゲットに分散して格納されます。

IOターゲットはファイルのデータ自体を格納するデータ領域と、ファイルタイプや更新時刻などを格納するメタデータ領域に分けられます。複数のIOターゲットを作成できますが、必ずデータ領域とメタデータ領域の個数は同数で一对一の関係です。

IOサーバに接続しているストレージの構成（プール、仮想ボリューム）からScaTeFSのメタデータ領域、データ領域のIOターゲットを設計します。

1つのIOサーバから2台のストレージの仮想ボリュームを使用するように設計します。

ScaTeFSのデータ領域

以下に設計例を記載します。

表 2-2 ScaTeFS データ領域のストレージ仮想ボリュームと LVM (VG) 構成

Storage	プール	仮想 ボリューム	iosv00		iosv01	
			VG	IOT	VG	IOT
1	DATA-POOL	DATA-LDEV1	vg_data01	0		
		DATA-LDEV2	vg_data02	1		
		DATA-LDEV3			vg_data07	6
		DATA-LDEV4			vg_data08	7
2	DATA-POOL	DATA-LDEV1			vg_data05	4
		DATA-LDEV2			vg_data06	5
		DATA-LDEV3	vg_data03	2		
		DATA-LDEV4	vg_data04	3		

- ScaTeFS のメタデータ領域
以下に設計例を記載します。

表 2-3 ScaTeFS メタデータ領域のストレージ仮想ボリュームと LVM (VG) 構成

Storage	プール	仮想 ボリューム	iosv00		iosv01	
			VG	IOT	VG	IOT
1	META-POOL	META-LDEV1	vg_ctrl01	0		
		META-LDEV2	vg_ctrl02	1		
		META-LDEV3			vg_ctrl07	6
		META-LDEV4			vg_ctrl08	7
2	META-POOL	META-LDEV1			vg_ctrl05	4
		META-LDEV2			vg_ctrl06	5
		META-LDEV3	vg_ctrl03	2		
		META-LDEV4	vg_ctrl04	3		

2.1.3 ネットワーク

IOサーバで使用するネットワーク情報 (IPアドレスなど) は事前に設計してください。

- 運用・管理用ネットワーク

メンテナンスのためクライアントやIOサーバにログインする場合や、ntpによる時刻同期等で使用するEthernetのネットワーク。

- ファイルシステム用ネットワーク

クライアントとIOサーバ間でデータやメタデータの転送を行うEthernetのネットワーク。

- IOサーバ間インタコネクト用ネットワーク

クラスタ構成のIOサーバでノード間の通信に使用します。クラスタ構成のIOサーバ間に閉じたネットワークであり、ご使用のネットワーク環境と衝突しないネットワークアドレスを設定してください。IPアドレスは2つ必要で、サーバID (SID) が偶数のIOサーバと奇数のIOサーバで異なるIPアドレスを設定してください。

2.1.4 CLUSTERPRO クラスタ構成情報の設計と作成

IOサーバの構築前にCLUSTERPROのクラスタ構成情報を作成します。作成したクラスタ構成情報は、「2.4.7 CLUSTERPRO」で使用します。

付録A「CLUSTERPROのクラスタ構成情報作成手順 (オフラインバージョン)」を参照してください。

2.2 ディストリビューションおよび各種ソフトウェアのインストール

IOサーバを利用するためのディストリビューションのインストールおよび各種ソフトウェアのインストールの手順を記載します。

本作業は、ストレージのセットアップの完了後に実施してください。以降の説明では 2 台の IOサーバをiosv00、iosv01と記載します。

2.2.1 Linux ディストリビューションのインストール

IOサーバがサポートするREHLを、下記ディストリビューションのインストール方法に従って、インストールします。

注意

「/ (ルート)」のデバイスタイプは「標準パーティション」を選択してください。

- ディストリビューションのインストール方法

Product Documentation for Red Hat Enterprise Linux 8

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8

2.2.2 RHEL のリポジトリの設定

IOサーバのソフトウェアをインストールするために、RHELに関する追加のパッケージをdnfコマンドでインストールできるように、RHELのyumリポジトリを設定します。RHELのyumリポジトリの設定については、RHELのインストール媒体を利用する方法とインターネット上の公式リポジトリを利用する方法があります。以下にRHELのインストール媒体を利用する方法を記載します。

(1) ディストリビューションのインストール媒体 (DVD) のマウント

RHEL DVD(ISOファイル)をダウンロードします。マウントポイントのディレクトリを作成してマウントします。もし、常時マウントする場合は、/etc/fstabへ記載します。

```
# mkdir /mnt/rhdvd
# mount -t iso9660 -o loop /root/rhel-8.6-x86_64-dvd.iso /mnt/rhdvd
```

(2) yumリポジトリの設定

ディストリビューションのインストール媒体 (DVD) からyum/dnfコマンドでパッケージをインストールできるようにするため、リポジトリ設定ファイルを作成します。

/etc/yum.repos.d/rhel8.6_dvd-Base.repoファイルの内容

```
[rhel86-baseos]
name=rhel86-baseos
baseurl=file:///mnt/rhdvd/BaseOS/
enabled=1
gpgcheck=1
```

/etc/yum.repos.d/rhel8.6_dvd-AppStream.repoファイルの内容

```
[rhel86-appstream]
name=rhel86-appstream
baseurl=file:///mnt/rhdvd/AppStream/
enabled=1
gpgcheck=1
```

yumリポジトリのキャッシュをクリアしてyumリポジトリが表示されることを確認します。

```
# yum clean all
# yum repoinfo rhel86-baseos
```

```
# yum repoinfo rhel86-appstream
```

2.2.3 時刻同期

IOサーバ間の時刻を一致させるために、Chronyを使用して時刻同期の設定を行います。

時刻同期の設定はRHEL 8のマニュアルを参照してください。

2.2.4 HA Dynamic Link Manager

iStorage Vシリーズでは、マルチパス制御ソフトHDLMを使用するため、HDLMのインストールが必要です。詳細はiStorage Vシリーズのマニュアル「iStorage Vシリーズ HA Command Suite Dynamic Link Manager ユーザーズガイド (Linux®用)」の3.6. HDLMのインストールを参照してください。

また、ストレージの論理ボリュームのデバイス名は、2台のIOサーバから同じデバイス名で参照できるように設定してください。

2.2.5 CLUSTERPRO

下記の手順でCLUSTERPROパッケージをインストールしてください。手順の詳細は、マニュアル「CLUSTERPRO X for Linux インストール&設定ガイド」を参照してください。

(1) インストール

CLUSTERPROパッケージをインストールします。

```
# rpm -ivh clusterpro-<version>.<architecture>.rpm
```

(2) ライセンス登録

ライセンスファイルを指定してライセンス登録を行います。

```
# clplcncs -i licensefile
```

2.2.6 リモートシェル(rsh)

リモートシェル関連のrsh、rsh-serverのパッケージをインストールします。

EPEL (Extra Packages for Enterprise Linux) リポジトリからrsh、rsh-serverパッケージをダウンロードしてください。

(1) インストール

rsh、rsh-serverのパッケージをインストールします。

```
# dnf install rsh-server-0.17-94.el8.x86_64.rpm rsh-0.17-94.el8.x86_64.rpm
```

(2) 設定

rshのサーバ機能を有効にします。

```
# systemctl enable rsh.socket
```

rshサーバ機能の設定ファイルの[Socket]タブに「MaxConnections=10000」を追加します。

```
# vi /etc/systemd/system/sockets.target.wants/rsh.socket
# cat /etc/systemd/system/sockets.target.wants/rsh.socket
[Unit]
Description=Remote Shell Facilities Activation Socket

[Socket]
ListenStream=514
Accept=true
MaxConnections=10000 ★追加

[Install]
WantedBy=sockets.target
```

rshサーバを起動します。

```
# systemctl daemon-reload
# systemctl start rsh.socket
```

2.2.7 ScaTeFS/Server

すべてのIOサーバに、ScaTeFS/Serverパッケージをインストールしてください。以下にインストール手順を記載します。

(1) ScaTeFS/Serverパッケージの入手

「インターネット配信製品ダウンロードサービス」を利用してScaTeFS/Serverのパッケージを含むzipファイルをダウンロードしてください。ダウンロードしたら、インストール対象マシンに転送し、zipファイルを展開します。

```
# unzip ScaTeFS_AI_S-YYYYMMDD.zip
(YYYYMMDDは年月日)
```

(2) ScaTeFS/Serverパッケージのインストール

ScaTeFS/Serverパッケージをインストールします。

```
# dnf install scatefs-ai-server/el8/scatefs-ai-server/*.rpm
```

ファイルシステムの統計情報をリアルタイムに収集しモニタリングする機能を使用する場合、モニタリングのパッケージをインストールします。

```
# dnf install scatefs-ai-server/el8/scatefs-ai-server-monitoring/*.rpm
```

2.2.8 ルーティング

複数のサブネット環境の場合は、ルーティング関連パッケージをインストールします。単一のサブネットの場合は、インストールは不要です。

IOサーバとゲートウェイの往路と復路を同じにするため、NetworkManager-dispatcher-routing-rulesをインストールします。

(1) NetworkManager-dispatcher-routing-rulesのインストール

NetworkManager-dispatcher-routing-rulesをインストールします。

```
# dnf install NetworkManager-dispatcher-routing-rules
```

2.2.9 Zabbix/Agent

ファイルシステムの統計情報をリアルタイムに収集しモニタリングする機能を使用する場合、Zabbixコミュニティからソフトウェアを入手してZabbix/Agentのパッケージをインストールします。

(1) Zabbix/Agent のパッケージのインストール

Zabbix/Agent のパッケージをインストールします。

```
# dnf install zabbix-agent-<version>.el8.x86_64.rpm
```

2.3 セットアップ

IOサーバを利用するための設定を記載します。

2.3.1 GRUB2 の設定

/etc/default/grubのGRUB_CMDLINE_LINUXを編集してGRUBの設定を行います。

(1) /etc/default/grub の編集

- IPv6 の無効化
「ipv6.disable=1」を追加します。
- SELinux の無効化
「selinux=0」を追加します。
- crashkernel の変更
「crashkernel=auto」から「crashkernel=1024M」へ変更します。
※ディストリビューションのインストール時に変更可能です。
- rhgb の削除
「rhgb」を削除します。

設定例を記載します。

```
GRUB_CMDLINE_LINUX="crashkernel=1024M resume=UUID=XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXX  
XXXXXXXX quiet selinux=0 ipv6.disable=1"
```

(2) grub.cfg ファイルの再構築

grub2-mkconfig コマンドを実行して、grub.cfg ファイルを再構築します。再構築後はディストリビューションを再起動します。

```
# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg  
# reboot
```

(3) ディストリビューション再起動後の確認

ディストリビューション再起動後、catコマンドで/proc/cmdline ファイルの内容を表示して、設定した内容が表示されることを確認します。

表示例を記載します。

```
# cat /proc/cmdline  
BOOT_IMAGE=/vmlinuz-4.18.0-372.32.1.el8_6.x86_64 root=UUID=XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX ro crashkernel=1024M quiet selinux=0 ipv6.disable=1
```

2.3.2 ファイアウォール

IOサーバでファイアウォールを設定する場合は、IOサーバで使用するポートの許可設定が必要になります。下記の設定を行ってください。

(1) ファイアウォールの状態を確認

状態がrunningであることを確認します。

```
# firewall-cmd -state
```

自動起動設定がenabledであることを確認します。

```
# systemctl is-enabled firewalld.service
```

(2) 使用しないサービスを削除

dhcpcv6-client、cockpitサービスを削除します。

```
# firewall-cmd --list-all
# firewall-cmd --permanent --zone=public --remove-service=dhcv6-client
# firewall-cmd --permanent --zone=public --remove-service=cockpit
```

(3) ScaTeFSサービスを追加

ScaTeFSサービスを追加します。

```
# firewall-cmd --permanent --new-service-from-file=/etc/scatefs/fws/scatefs.xml
--name=scatefs
# firewall-cmd --permanent --add-service=scatefs --zone=public
```

(4) CLUSTERPROサービスを追加

CLUSTERPROサービスを追加します。

```
# /opt/nec/clusterpro/bin/clpfwctrl.sh --add --zone=public
```

(5) 追加したscatefs、clusterproサービスを確認

追加したscatefs、clusterproサービスが表示されることを確認します。

```
# firewall-cmd --reload
# firewall-cmd --list-all
```

2.3.3 ネットワーク

IOサーバでは複数のネットワークポートを使用します。それぞれについて、ネットワーク設計の通りにIPアドレスの設定等を行ってください。

2.3.3.1 ファイルシステムポート

ScaTeFS クライアントからのファイルアクセスに使用します。ファイルシステムポートで称するネットワークインタフェースには下記二つの IP アドレスを設定します。

- 実 IP アドレス
CLUSTERPRO のフローティング IP リソースのアドレス割り当ての条件で、実 IP アドレスを設定します。
- フローティング IP アドレス
CLUSTERPRO のフローティング IP リソースが起動時に IP アドレスを設定します。

以下に nmcli コマンドによるファイルシステムポートの設定例を記載します。

【例：ネットワークボンディングを使用しないケース】

ファイルシステムポート

ens2f0 : 172.16.6.6/25

ens2f0の設定
<pre># nmcli connection modify ens2f0 connection.autoconnect yes # nmcli connection modify ens2f0 ipv4.never-default true # nmcli connection modify ens2f0 ipv4.method manual ipv4.address "172.16.6.6/25" # nmcli connection modify ens2f0 ipv6.method ignore # nmcli connection up ens2f0 # ip a show dev ens2f0</pre>

【例：ネットワークボンディングを使用するケース】

ファイルシステムポート

ens2f0, ens2f1, bond0 : 172.16.6.6/25

ens2f0, ens2f1, bond0の設定

```
# nmcli connection add type bond con-name bond0 ifname bond0
# nmcli connection add type ethernet autoconnect yes ifname ens2f0 master
bond0
# nmcli connection add type ethernet autoconnect yes ifname ens2f1 master
bond0
# nmcli connection modify bond0 ipv4.never-default true
# nmcli connection modify bond0 ipv4.method disabled ipv6.method ignore
# nmcli connection modify bond0 ipv4.method manual ipv4.address
"172.16.6.6/25"
# nmcli connection modify bond0 +bond.options
mode=802.3ad,miimon=100,xmit_hash_policy=layer2+3
# nmcli connection up bond0
# ip a show dev bond0
```

2.3.3.2 IO サーバ間インタコネク用ポート

ペアの IO サーバ間の通信に使用します。ペアの IO サーバ間に閉じているネットワークのため、ご使用のネットワーク環境と衝突しないネットワークアドレスを設定してください。

以下に nmcli コマンドによる IO サーバ間インタコネク用ポートの設定例を記載します。

【例：IO サーバ間インタコネク用ポート】

IO サーバ間インタコネク用ポート

IO サーバ0

ens1f0np0, ens1f1np1, bond2 : 10.2.0.10

ens1f0np0, ens1f1np1, bond2 の設定

```
# nmcli connection add type bond con-name bond0 ifname bond2
# nmcli connection add type ethernet autoconnect yes ifname ens1f0np0
master bond2
# nmcli connection add type ethernet autoconnect yes ifname ens1f1np1
master bond2
# nmcli connection modify bond2 ipv4.never-default true
# nmcli connection modify bond2 ipv6.method ignore
# nmcli connection modify bond2 +bond.options
mode=802.3ad,miimon=100,xmit_hash_policy=layer2+3

・ IO サーバ0
# nmcli connection modify bond2 ipv4.method manual ipv4.address
"10.2.0.10/24"
# nmcli connection up bond2

・ IO サーバ1
```

ens1f0np0, ens1f1np1, bond2 の設定

```
# nmcli connection modify bond2 ipv4.method manual ipv4.address
"10.2.0.11/24"
# nmcli connection up bond2
```

2.3.3.3 ファイルシステムポートのルーティング設定

IO サーバとゲートウェイの往路と復路を同じにするため、ルーティングを設定します。

以下にファイルシステムポートのルーティング設定例を記載します。

【例：ネットワークボンディングを使用しないケース】

ファイルシステムポート

ens2f0 : 172.16.6.6/25

ゲートウェイ

172.16.6.126

(1) Ip rule の設定

/etc/sysconfig/network-scripts/rule-ens2f0 ファイルを作成します。

例ではテーブル ID は 200 とします。

rule-ens2f0 の設定例

```
table 200 from 172.16.6.0/25
```

(2) ルーティングと ip route の設定

/opt/scatefs/script/routeadd.sh ファイルにルーティングを設定します。

routeadd.sh は CLUSTERPRO がフローティング IP アドレスを設定した後に動作するスクリプトです。

routeadd.sh の設定例

```
#!/bin/sh

# routing add script

## ip route
ip route add table 200 172.16.6.0/25 dev bond0.12 proto kernel src
172.16.6.62
ip route add table 200 default via 172.16.6.126
```

2.3.4 ファイルシステム管理アカウント

IOサーバにおけるファイルシステムの管理・運用の操作は、fsadminアカウントにて行います。fsadminアカウント自体はscatefs-ai-srvパッケージをインストールすることによりIOサーバに作成されますが、このfsadminアカウントでIOサーバ間のリモート実行が可能ないように設定を行います。

```
# su - fsadmin
-bash-4.1$ vi .rhosts
-bash-4.1$ chmod 600 .rhosts
-bash-4.1$ exit
#
```

fsadminの.rhostsファイルに全IOサーバのIPアドレスを記載してください。運用・管理ポートとファイルシステムポートの両方のアドレスを記載してください。

設定後は、IOサーバ間でfsadminによるリモート実行が可能であることを確認してください。

ファイルシステムポートのリモートシェル実行は、CLUSTERPROのクラスタ状態が正常のときに確認します。

```
# su - fsadmin
-bash-4.1$ rsh iosv01 hostname
iosv01
-bash-4.1$
```

2.3.5 内蔵ディスク (ScaTeFS 用) の設定

/homeパーティションへScaTeFS用ディレクトリを作成します。

```
# mkdir -p /home/scatefs/ssd
# mkdir -p /home/scatefs/core
# ln -s /home/scatefs/ssd /mnt
# ln -s /home/scatefs/core /mnt
```

2.3.6 カーネルパラメータ

/etc/sysctl.confファイルの末尾に以下を追加します。

```
# ScaTeFS
vm.dirty_writeback_centisecs = 2
vm.dirty_expire_centisecs = 10
```

```
vm.swappiness = 0
net.core.somaxconn = 4000
net.ipv4.ip_local_reserved_ports = 50000-50015
kernel.core_pattern = /mnt/core/core.%e
kernel.core_uses_pid = 0
kernel.unknown_nmi_panic = 1
kernel.panic_on_unrecovered_nmi = 1
```

上記設定を反映させるため、sysctlコマンドを実行します。

```
# sysctl -p
```

2.3.7 systemd の設定

systemdのログレベルをnoticeへ変更します。

/etc/systemd/system.confファイルへ「LogLevel=notice」を追加します。

```
# vi /etc/systemd/system.conf
-----
LogLevel=notice
-----
```

ディストリビューション再起動後、ログレベルを確認します。

```
# systemctl -pLogLevel show
LogLevel=notice
```

2.3.8 sysstat の設定

IOサーバのシステムリソース情報を採取するため、sysstatをインストールして設定します。

(1) sysstat パッケージをインストール

sysstat パッケージをインストールします。

```
# dnf install sysstat
```

(2) 情報採取間隔を 1 分に変更

sysstat-collect.timer の設定を変更します。

```
# EDITOR=vim systemctl edit sysstat-collect.timer
```

```
# systemctl cat sysstat-collect.timer
```

設定内奥

下記内容をコピーしてペーストします。

```
[Unit]
Description=Run system activity accounting tool every 1 minute

[Timer]
OnCalendar=
OnCalendar=*:00/1
```

(3) sysstat-collect.timer を開始

sysstat-collect.timer を開始します。

```
# systemctl start sysstat-collect.timer
# systemctl status sysstat-collect.timer
```

2.3.9 crashdump の設定

IOサーバでcrashdumpを取得できるように設定します。

詳細は「3.1.2 カーネルクラッシュダンプ採取の有効化」を参照してください。

2.4 ScaTeFSのセットアップ

ScaTeFSを利用するための設定を記載します。

2.4.1 IO サーバの登録

scatefs_addiosコマンドを実行して、すべてのIOサーバをScaTeFSに登録します。

以下にscatefs_addiosコマンドに指定するデータファイルの設定項目を記載します。

表 2-4 scatefs_addios の設定項目

設定項目	説明	必須
ipaddr	運用・管理ポートのIPアドレス	○
fipaddr	ファイルシステムポートのIPアドレス 複数登録する場合はスペース区切りで指定します	○
inipaddr	IOサーバ間インタコネク用ポートのIPアドレス	○
cport	クライアント接続ポート番号 デフォルト値50000から変更しない場合は省略可	-
sport	サーバ間通信接続ポート番号 デフォルト値50001から変更しない場合は省略可	-
cdport	データ転送用クライアント接続ポート番号 デフォルト値50002から変更しない場合は省略可	-
iftypes	fipaddr のインターフェースタイプ Ethernetの場合は 1 を指定します fipaddrを複数登録する場合は、fipaddrと同じ数だけ スペース区切りで指定します	○

(1) データファイルの作成

データファイルを作成します。

以下にデータファイルの例を記載します。

- ファイルシステムポートに Ethernet を使用する場合

```
-bash-4.1$ cat datafile1
# IOS#0の設定
ipaddr 10.0.0.1
fipaddr 10.0.1.1 10.0.2.1
inipaddr 10.2.0.10
iftypes 1 1

# IOS#1の設定
ipaddr 10.0.0.2
```

```
fipaddr 10.0.1.2 10.0.2.2
inipaddr 10.2.0.11
iftypes 1 1
```

(2) scatefs_addios の実行

ルート IO サーバ (1 番目の IO サーバ) で scatefs_addios コマンドを実行します。

1 回の実行によりすべての IO サーバに情報を登録します。

```
# su - fsadmin
-bash-4.1$ scatefs_addios -f datafile1
```

設定した情報は scatefs_detail -s コマンドで確認できます。

```
-bash-4.1$ scatefs_detail -s
```

IOSID	MATE	IP[0]	IPCNT	FIP[0]	FIPCNT	IOTCNT	FSCNT
0	1	10.0.0.1	1	10.0.1.1	2	0	0
1	0	10.0.0.2	1	10.0.1.2	2	0	0

```
ALL:2 CAPACITY:256
```

2.4.2 HDLM デバイス名の確認

ストレージの仮想ボリュームの認識順序によって、IOサーバ間でHDLMデバイス名が異なる場合があります。IOサーバ間でHDLMデバイス名が異なる場合は、IOサーバ間で同じHDLMデバイス名になるように変更してください。詳細はiStorage Vシリーズのマニュアル「iStorage Vシリーズ HA Command Suite Dynamic Link Manager ユーザーズガイド (Linux®用)」の「4.6.5. HDLMデバイス名の変更」を参照してください。

2.4.3 パーティションの作成

メタデータ領域用仮想ボリュームに以下のパーティションを作成します。

(1) CLUSTERPRO のディスクハートビート用のパーティション

メタデータ領域用の各仮想ボリュームの先頭に 16MB 程度のパーティションを作成します。

(2) メタデータ領域用のパーティション

メタデータ領域用仮想ボリュームが複数の場合、残りの容量を一つのパーティションとして作成します。

また、メタデータ領域用仮想ボリュームを一つにすることができます。その場合、データ領域用仮想ボリューム数分のパーティションを作成します。パーティションの容量が均等になるように作成します。

以下にメタデータ領域用仮想ボリュームが複数の場合の作成例を記載します。

```
# parted /dev/sddlmaa
GNU Parted 3.2
/dev/sddlmaa を使用
GNU Parted へようこそ！ コマンド一覧を見るには 'help' と入力してください。
(parted) mklabel gpt
(parted) mkpart primary ext4 0% 16MB
(parted) mkpart primary ext4 16MB 100%
(parted) print
モデル: 不明 (unknown)
ディスク /dev/sddlmaa: 752GB
セクタサイズ (論理/物理): 512B/512B
パーティションテーブル: gpt
ディスクフラグ:

番号  開始      終了      サイズ  ファイルシステム  名前      フラグ
  1    1049kB    15.7MB    14.7MB  ext4              primary
  2    15.7MB   752GB     752GB   ext4              primary

(parted) q
通知: 必要であれば /etc/fstab を更新するのを忘れないようにしてください。

#
```

2.4.4 LVM デバイスの作成

LVM設計内容からLVMデバイスを作成します。LVMデバイスの作成はHDLMのデバイスファイル (/dev/sddlXX) を使用します。LVMデバイスを作成した後にLVMデバイスを認識させるため、IOサーバでディストリビューションの再起動を行います。ディストリビューションの再起動後に、作成したLVMデバイス (デバイスファイル) が存在することを確認します。

- ScaTeFS のデータ領域の LVM デバイスの作成例

[PV]

iosv00

```
# pvcreate /dev/sddlmaa
# pvcreate /dev/sddlmaab
# pvcreate /dev/sddlmaak
# pvcreate /dev/sddlmal
```

iosv01

```
# pvcreate /dev/sddlmai
# pvcreate /dev/sddlmaaj
# pvcreate /dev/sddlmac
# pvcreate /dev/sddlmad
```

[VG]

iosv00

```
# vgcreate vg_data01 /dev/sddlmaa
# vgcreate vg_data02 /dev/sddlmaab
# vgcreate vg_data03 /dev/sddlmaak
# vgcreate vg_data04 /dev/sddlmal
```

iosv01

```
# vgcreate vg_data05 /dev/sddlmai
# vgcreate vg_data06 /dev/sddlmaaj
# vgcreate vg_data07 /dev/sddlmac
# vgcreate vg_data08 /dev/sddlmad
```

[LV]

iosv00

```
# lvcreate -l 100%free -r none -n lv_data01 vg_data01
# lvcreate -l 100%free -r none -n lv_data02 vg_data02
# lvcreate -l 100%free -r none -n lv_data03 vg_data03
# lvcreate -l 100%free -r none -n lv_data04 vg_data04
```

iosv01

```
# lvcreate -l 100%free -r none -n lv_data05 vg_data05
# lvcreate -l 100%free -r none -n lv_data06 vg_data06
# lvcreate -l 100%free -r none -n lv_data07 vg_data07
# lvcreate -l 100%free -r none -n lv_data08 vg_data08
```

- ScaTeFS のメタデータ領域の LVM デバイスの作成例

[LV]

iosv00

```
# pvcreate /dev/sddlmae2
# pvcreate /dev/sddlmaf2
# pvcreate /dev/sddlmao2
# pvcreate /dev/sddlmap2
```

iosv01

```
# pvcreate /dev/sddlma2
# pvcreate /dev/sddlman2
# pvcreate /dev/sddlma2
# pvcreate /dev/sddlma2
```

[VG]

iosv00

```
# vgcreate vg_ctrl01 /dev/sddlmae2
# vgcreate vg_ctrl02 /dev/sddlmaf2
# vgcreate vg_ctrl03 /dev/sddlmao2
# vgcreate vg_ctrl04 /dev/sddlmap2
```

iosv01

```
# vgcreate vg_ctrl05 /dev/sddlma2
# vgcreate vg_ctrl06 /dev/sddlman2
# vgcreate vg_ctrl07 /dev/sddlma2
# vgcreate vg_ctrl08 /dev/sddlma2
```

[LV]

iosv00

```
# lvcreate -l 100%free -r none -n lv_ctrl01 vg_ctrl01
# lvcreate -l 100%free -r none -n lv_ctrl02 vg_ctrl02
# lvcreate -l 100%free -r none -n lv_ctrl03 vg_ctrl03
# lvcreate -l 100%free -r none -n lv_ctrl04 vg_ctrl04
```

iosv01

```
# lvcreate -l 100%free -r none -n lv_ctrl05 vg_ctrl05
# lvcreate -l 100%free -r none -n lv_ctrl06 vg_ctrl06
# lvcreate -l 100%free -r none -n lv_ctrl07 vg_ctrl07
# lvcreate -l 100%free -r none -n lv_ctrl08 vg_ctrl08
```

2.4.5 IO ターゲットの登録

scatefs_addiotコマンドを実行して、作成したLVM論理ボリューム (LV) をIOターゲットとしてScaTeFSに登録します。

以下にscatefs_addiotコマンドに指定するデータファイルの設定項目を記載します。

表 2-5 scatefs_addiot の設定項目

設定項目	説明	必須
iosid	IO サーバのサーバ ID (SID)	○
data	IO ターゲットのデータ領域のデバイス名	○
ctrl	IO ターゲットのメタデータ領域のデバイス名	○

(1) データファイルの作成

データファイルを作成します。

以下にデータファイルの例を記載します。

```
-bash-4.1$ cat datafile2
iosid 0
data /dev/vg_data01/lv_data01
ctrl /dev/vg_ctrl01/lv_ctrl01
data /dev/vg_data02/lv_data02
ctrl /dev/vg_ctrl02/lv_ctrl02
data /dev/vg_data03/lv_data03
ctrl /dev/vg_ctrl03/lv_ctrl03
data /dev/vg_data04/lv_data04
ctrl /dev/vg_ctrl04/lv_ctrl04

iosid 1
data /dev/vg_data05/lv_data05
```

```
ctrl    /dev/vg_ctrl05/lv_ctrl05
data    /dev/vg_data06/lv_data06
ctrl    /dev/vg_ctrl06/lv_ctrl06
data    /dev/vg_data07/lv_data07
ctrl    /dev/vg_ctrl07/lv_ctrl07
data    /dev/vg_data08/lv_data08
ctrl    /dev/vg_ctrl08/lv_ctrl08
```

(2) scatefs_addiot の実行

ルート IO サーバ (1 番目の IO サーバ) で scatefs_addiot コマンドを実行します。

1 回の実行によりすべての IO サーバに情報を登録します。

```
# su - fsadmin
-bash-4.1$ scatefs_addiot -f datafile2
```

設定した情報は scatefs_detail -t コマンドで確認できます。

```
$ scatefs_detail -t
```

IOTID	IOS	FS:SG
0	0	none:none
1	0	none:none
2	0	none:none
3	0	none:none
4	1	none:none
5	1	none:none
6	1	none:none
7	1	none:none

```
ALL:8 USED:0 CAPACITY:16384
```

2.4.6 ScaTeFS のファイルシステム作成

scatefs_mkfs コマンドを実行して、ScaTeFS のファイルシステムを作成します。

以下に scatefs_mkfs コマンドに指定するデータファイルの設定項目を記載します。

表 2-6 scatefs_mkfs の設定項目

設定項目	説明	必須
name	ファイルシステム名 クライアントノードからマウントする際に指定します 31 文字まで指定可能です	○
iotid	IOターゲットのID 下記で設計する「IO ターゲットを使用する順序」を指定します	○
data_fstype	データ領域のファイルシステムタイプはxfsを指定します 省略した場合のデフォルトは ext4 です	○

(1) IO ターゲットを使用する順序

IO サーバとストレージの負荷を分散するように IO ターゲットを使用する順序を設計します。
IO サーバから見て 2 台のストレージの論理ボリュームを交互に使用します。
以下に設計例を記載します。

表 2-7 IO ターゲットを使用する順序

IO サーバ	IO ターゲットを使用する順序
iosv00	0 3 1 2
iosv01	4 7 5 6

データファイルの設定項目の iotid の設定値は、上記 iosv00、iosv01 の IO ターゲットを使用する順序を続けて記載します。

表 2-8 iotid の設定値

設定項目	設定値
iotid	0 3 1 2 4 7 5 6

(2) データファイルの作成

データファイルを作成します。
以下にデータファイルの例を記載します。

```
-bash-4.1$ cat datafile3
name          scatefs00
iotid         0 3 1 2 4 7 5 6
```

```
datafstype xfs
```

(3) scatefs_mkfs の実行

ルート IO サーバ(1 番目の IO サーバ)で scatefs_mkfs コマンドを実行します。

1 回の実行によりすべての IO サーバに情報を登録します。

scatefs_mkfs により、各 IO サーバで IO ターゲットの mkfs が行われ、ローカルにマウントされます。さらに ScaTeFS としてのフォーマットが行われます。

```
# su - fsadmin
-bash-4.1$ scatefs_addiot -f datafile2
```

設定した情報は scatefs_detail -f コマンドで確認できます。

```
-bash-4.1$ scatefs_detail -f
```

FSID	NAME	ROOTIOS	IOSCNT	IOTCNT	SGCNT	VERSION
0	scatefs00	0	2	8	1	0x00010000

```
ALL:1 CAPACITY:32
```

2.4.7 ScaTeFS ライセンス

ScaTeFSのライセンス登録を行います。

手順の詳細については、「NEC Scalable Technology File System for AI (ScaTeFS for AI) ソフトウェアライセンス管理説明書」をご覧ください。

2.4.8 CLUSTERPRO

これまで構築しました各リソースの管理をCLUSTERPROで行うため、CLUSTERPROの設定を行います。CLUSTERPROの設定はCluster WebUIを使用するため、IOサーバへネットワーク接続できる作業用PCが必要となります。

※Cluster WebUIの操作方法と各リソースの設定については、マニュアル「CLUSTERPRO X for Linux リファレンスガイド」を参照してください。

(1) クラスタ構成情報ファイルを作業用 PC へ転送

「2.1.4 CLUSTERPRO クラスタ構成情報の設計と作成」で作成したクラスタ構成情報ファイルを作業用 PC へ転送します。

(2) Cluster WebUI の起動

Web ブラウザを起動してアドレスバーに IO サーバの IP アドレス(管理用)とポート番号を入力します。

※もし接続できない場合、ペアの IO サーバの IP アドレスを指定します。

`http://10.0.0.1:29003`

(3) クラスタ構成情報ファイルのインポート

ツールバーのドロップダウンメニューで[設定モード] を選択します。「設定のインポート」をクリックして、クラスタ構成情報ファイルをインポートします。

(4) クラスタ構成情報の変更

ディスクハートビート用デバイスを設定します。

cluster のプロパティを開き、インタコネクトタブをクリックします。

以下のハートビート I/F の設定を変更します。

優先度 2 : 種別 (DISK)

CLUSTERPRO のハートビート領域用のパーティションのデバイス名へ変更します。

(5) クラスタ構成情報の設定反映

Cluster WebUI の「設定の反映」をクリックして、設定反映を行います。

設定反映後にクラスタ構成の IO サーバで OS 再起動を行います。OS 起動後に `clpstat` コマンドを実行して、クラスタの状態を確認します。

2.5 アップデート

ScaTeFS/Serverパッケージのアップデートについて、以下の方法があります。

- システム運用を停止して行うアップデート
- システム運用を継続して行うアップデート(無停止アップデート)

以下にScaTeFS/Serverパッケージをアップデートする手順を記載します。

2.5.1 システム運用を停止して行うアップデート

システム運用を停止してScaTeFS/Serverパッケージをアップデートする手順を記載します。

すべてのIOサーバに、ScaTeFS/Serverパッケージをアップデートしてください。

(1) ScaTeFS/Serverパッケージの入手

「NECサポートポータル」を利用してScaTeFS/Serverのパッケージを含むzipファイルをダウンロードしてください。ダウンロードしたら、インストール対象マシンに転送し、zipファイルを展開します。

```
# unzip ScaTeFS_AI_Server_N.N.zip  
(Nは数字)
```

(2) IOサーバデーモンの停止

CLUSTERPROのclprscコマンドでIOサーバデーモンを停止します。

```
# clprsc -t exec1  
# clprsc -t exec2
```

(3) ScaTeFS/Serverパッケージのアップデート

ScaTeFS/Serverパッケージをアップデートします。

```
# dnf update scatefs-ai-server/el8/scatefs-ai-server/*.rpm
```

ファイルシステムの統計情報をリアルタイムに収集しモニタリングする機能を使用する場合、Zabbix/Agentサービスを停止してモニタリングのパッケージをアップデートします。

```
# systemctl stop zabbix-agent  
# dnf update scatefs-ai-server/el8/scatefs-ai-server-monitoring/*.rpm  
# systemctl start zabbix-agent
```

(4) IOサーバデーモンの起動

CLUSTERPROのclprscコマンドでIOサーバデーモンを起動します。

```
# clprsc -s exec1  
# clprsc -s exec2
```

2.5.2 システム運用を継続して行うアップデート（無停止アップデート）

システム運用を継続してScaTeFS/Serverパッケージをアップデート（無停止アップデート）する手順を記載します。

アップデートは、各IOサーバに管理者権限（root）でログインし、以下の手順で実施してください。

なお、アップデート中は当該IOサーバへのIOが最大で3.5分間遅延します。このため、複数のIOサーバを連続してアップデートした場合、その台数分IOが遅延する可能性がありますので、運用に影響

が出ないように十分な間隔をあけてください（最低8分は間隔をあけてください）。

(1) 準備

「NECサポートポータル」を利用してScaTeFS/Serverのパッケージを含むzipファイルをダウンロードしてください。ダウンロードしたら、インストール対象マシンに転送し、zipファイルを展開します。

```
# unzip ScaTeFS_AI_Server_N.N.zip
(Nは数字)
```

(2) 環境の確認

clpstatコマンドでクラスタ状態を確認してください。詳細は「NEC Scalable Technology File System for AI (ScaTeFS for AI) ユーザーズガイド」の「3.3.2 IOサーバの起動と停止」の起動確認を参照してください。

問題がある場合、該当リソースに何らかの障害が発生していますので、無停止アップデートは実施せず障害を解消してください。

(3) アップデート

- ① IOサーバデーモンが実行されている状態でパッケージを適用します。

```
# dnf update scatefs-ai-server/el8/scatefs-ai-server/*.rpm
```

- ② 以下のコマンドを実行します。

```
# /opt/scatefs/sbin/scatefs_restart
```

(4) 結果確認

- ②のコマンドの実行結果が正常終了 (0) していればアップデートは完了です。

異常終了 (1) している場合、clpstat コマンドで環境を確認し、下記のそれぞれの状況に応じた対応を実施した上で、サポート部門へ連絡してください。

- フェイルオーバーしている場合、以下のコマンドでアップデート前のパッケージに戻した上で、テイクバックを実施してください。

以前に適用したScaTeFS/Serverパッケージファイルを指定して元に戻します。

```
# dnf downgrade アップデート前のScaTeFS/Serverパッケージファイル
```

- その他

アップデート前のパッケージに戻した上で、scatefs_restartコマンドを再実行してください。

第3章 Linux クライアントのセットアップと運用

3.1 クライアントのインストール

3.1.1 Linux ディストリビューションのインストール

ScaTeFSクライアントで使用できるLinuxディストリビューションとカーネルバージョンは、以下のとおりです。

表 3-1 使用できる Linux ディストリビューションとカーネルバージョン

Linuxディストリビューション	カーネルバージョン
RHEL 8.6	4.18.0-372.32.1.el8_6.x86_64

Linuxディストリビューションは、そのインストール方法にしたがってインストールしてください。

また、既にインストールされているカーネルバージョンがScaTeFSクライアントのパッケージを適用できるカーネルバージョンと異なる場合、ScaTeFSクライアントのパッケージを適用できるバージョンのカーネルをインストールし、インストール対象マシンを再起動してください。

以降の作業は、ScaTeFSクライアントのパッケージを適用できるカーネルバージョンで起動した環境で行います。

3.1.2 カーネルクラッシュダンプ採取の有効化

カーネルクラッシュやシステムストールの発生時に原因を解析できるように、クラッシュダンプ採取を有効にすることを強く推奨します。

RHEL 8の「カーネルの管理、監視、および更新」のマニュアルの「第14章 kdump のインストール」から「第20章 コアダンプの分析」までの章を参照し、以下の作業を行ってください。

- kdumpのインストールと設定
- kdump設定のテスト（実際にカーネルクラッシュダンプが採取できることの確認）
- crashユーティリティのインストール

3.1.3 ScaTeFS クライアントのパッケージを含む zip ファイルの入手

- (1) 「インターネット配信製品ダウンロードサービス」を利用して ScaTeFS クライアントのパッケージを含む zip ファイルをダウンロードしてください。
- (2) ダウンロードした zip ファイルをインストール対象マシンに転送し、zip ファイルを展開しま

す。

```
# unzip ScaTeFS_AI_C-YYYYMMDD.zip  
(YYYYMMDDは年月日)
```

3.1.4 ScaTeFS クライアントのパッケージのインストール

以下のコマンドでインストールします。

- RHEL 8.6の場合

```
# dnf install scatefs-ai-client/el8.6/scatefs-ai-client/*.rpm
```

3.1.5 インストール対象マシンの再起動

以下のコマンドで再起動します。

```
# reboot
```

3.2 クライアントのセットアップ

3.2.1 ルーティング設定

ScaTeFSクライアントは、IOサーバのネットワークインターフェースと通信を行います。IOサーバが複数のネットワークインターフェースをもつ場合、それぞれのネットワークインターフェースで通信できるように静的なルーティング設定を追加してください。

現在のルーティング設定は以下の方法で確認できます。ルーティングの設定に関する詳細はRHELのマニュアル等を参照してください。

- ipコマンドでルーティングテーブルを表示

```
# ip route
```

- netstatコマンドでルーティングテーブルを表示

```
# netstat -r
```

ScaTeFSクライアントのルーティング設定が正しくない場合、以下のような現象が発生します。

- mountの応答が返って来ない
ルートIOサーバに対するルーティング設定を確認してください。
- mountはできたが、その後のScaTeFSへのアクセスで応答が返らないことがある
IOサーバが複数のネットワークインターフェースをもつ場合、それらのネットワークインターフェースとの接続の一部が張れていない可能性があります。
Linuxではpingコマンドの-Iオプションを使うと送信元のネットワークインターフェースを指定できます。ScaTeFSクライアントで使うネットワークインターフェースを指定して、IOサーバの各ネットワークインターフェースと通信できるか確認します。
[IOサーバが2つのネットワークインターフェースをもつ場合の例]

```
# ping "IOサーバのネットワークインターフェース1のIPアドレス" -I "クライアントのインターフェース名"
# ping "IOサーバのネットワークインターフェース2のIPアドレス" -I "クライアントのインターフェース名"
```

3.2.2 マウント方法

mountコマンドを使ってファイルシステムをマウントします。

以下に、ルートIOサーバ"iosv00"のファイルシステム"scatefs00"を/mnt/scatefsにマウントする例を示します。

```
# mount -t scatefs -o rsize=4194304,wsiz=4194304 iosv00:scatefs00 /mnt/scatefs
```

マウントオプションのrsizeとwsizは、クライアントとIOサーバの間でファイルのデータを入出力する際の転送サイズを表します。ともに既定値は1MBですが、2MB、または4MBとした方が性能は向上します。

マウントオプションの詳細については、『NEC Scalable Technology File System for AI (ScaTeFS for AI) ユーザーズガイド』の「4.1.1.1 scatefs」を参照してください。

ファイルシステムに関する情報を/etc/fstabに記述し、Linuxマシンの起動時にファイルシステムを自動的にマウントする場合、マウントオプションに_netdevを記述してください。本オプションを記述しない場合、RHEL 8ではLinuxマシンの起動時にファイルシステムのマウントに失敗し、緊急モードのログインプロンプトがコンソールに表示されます。この場合、マウントオプションに_netdevを追加し再起動してください。以下に/etc/fstabの記述例を示します。

```
iosv00:scatefs00 /mnt/scatefs scatefs _netdev,rsize=4194304,wsiz=4194304 0 0
```

マウントオプションとしてSELinuxのコンテキストが指定されなかった場合、既定値として context="system_u:object_r:nfs_t:s0" が使用されます。他のコンテキストを使用したい場合は、マウントオプションでコンテキストを指定してください。

3.2.3 全 IO サーバとのコネクション確立の確認

マウント後、全IOサーバとのコネクションが確立できるかどうか、IO確認を行います。

ScaTeFSは、同一ディレクトリにファイルを作成すると、各IOサーバにラウンドロビンで分散しますので、それを利用して確認します。

以下は、IOサーバ2台で実施する場合の例です。マウントポイントとループ回数は適宜変更してください。ループ回数はIOサーバ数以上とします。

```
# for N in {1..2}; do dd if=/dev/zero of=/mnt/scatefs/testfile${N} bs=10M count=1; done;
```

現在確立されているコネクションは、以下の方法で確認できます。

- ss コマンドを用いて確認する場合

```
# ss -nt | egrep 'State|:5000'
```

- netstatコマンドを用いて確認する場合

```
# netstat -n | egrep 'Local|:5000' | sort
```

表示されるローカル側アドレスとリモート側アドレスを見て、意図したネットワークインターフェースが利用されているか確認してください。特に、ローカルアドレスが意図したネットワークインターフェースになっていることを確認します。

以下は、ネットワークインターフェースが2つあるIOサーバを2台用いて ScaTeFS を構築したときの例です。

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	172.28.134.43:869	172.16.6.5:50000	ESTABLISHED
tcp	0	0	172.28.134.43:869	172.16.6.5:50002	ESTABLISHED
tcp	0	0	172.28.134.43:869	172.16.6.6:50000	ESTABLISHED
tcp	0	0	172.28.134.43:869	172.16.6.6:50002	ESTABLISHED
tcp	0	0	172.28.134.43:869	172.16.7.5:50000	ESTABLISHED
tcp	0	0	172.28.134.43:869	172.16.7.5:50002	ESTABLISHED
tcp	0	0	172.28.134.43:869	172.16.7.6:50000	ESTABLISHED
tcp	0	0	172.28.134.43:869	172.16.7.6:50002	ESTABLISHED

ScaTeFS では、IO サーバのネットワークインターフェース 1 つに対し、ポート番号 50000 (メタデータ通信用) と 50002 (データ通信用) の 2 つのコネクションを作成します。

また、コネクションは IO サーバとの通信が必要になった時点で作成するため、IO がまだあまりされていない段階では、コネクションは一部のみ張られた状態になります。

3.2.4 アンマウント方法

umount コマンドを使ってファイルシステムをアンマウントします。

以下に、/mnt/scatefs にマウントされているファイルシステムをアンマウントする例を示します。

```
# umount /mnt/scatefs
```

IO サーバとの通信が不通となった場合、-f オプションを使用することによりファイルシステムを強制的にアンマウントすることができます。以下に、/mnt/scatefs にマウントされているファイルシステムを強制的にアンマウントする例を示します。

```
# umount -f /mnt/scatefs
```

3.3 クライアントのアップデート

3.3.1 ファイルシステムのアンマウントと scatefs-client サービスの停止

- (1) アップデート対象マシンで ScaTeFS のファイルシステムを NFS でエクスポートしている場合、そのファイルシステムをマウントしている全ての NFS クライアントからそのファイルシステムをアンマウントしてください。アンマウントが完了したら (2) へ進んでください。

アップデート対象マシンで ScaTeFS のファイルシステムを NFS でエクスポートしていない場合は、(3) に進んでください。

- (2) アップデート対象マシンで nfs-server サービスを停止します。

```
# systemctl stop nfs-server
```

- (3) アップデート対象マシンで ScaTeFS のファイルシステムを Samba で共有として公開している場合、その共有を使用している全ての Windows 等の CIFS クライアントでその共有へのアクセスを止めてください。アクセスの停止が完了したら (4) へ進んでください。

アップデート対象マシンで ScaTeFS のファイルシステムを Samba で共有として公開していない場合は、(5) に進んでください。

(4) アップデート対象マシンで smb サービスと nmb サービスを停止します。

```
# systemctl stop smb
# systemctl stop nmb
```

(5) アップデート対象マシンで ScaTeFS のファイルシステムをアンマウントします。

```
# umount -a -t scatefs
```

(6) アップデート対象マシンで scatefs-client サービスを停止します。

```
# systemctl stop scatefs-client
```

3.3.2 ScaTeFS クライアントのパッケージのアンインストール

以下のコマンドでパッケージをアンインストールします。

```
# dnf remove '*scatefs-ai-client-*
```

3.3.3 カーネルのアップデート (選択)

アップデート対象マシンのカーネルバージョンが、ScaTeFS クライアントのパッケージを適用できるカーネルバージョンと異なる場合、ScaTeFS クライアントのパッケージを適用できるバージョンのカーネルをインストールし、アップデート対象マシンを再起動してください。

以降の作業は、ScaTeFS クライアントのパッケージを適用できるカーネルバージョンで起動した環境で行います。

3.3.4 ScaTeFS クライアントのパッケージを含む zip ファイルの入手

(1) 「NEC サポートポータル」を利用して ScaTeFS クライアントのパッケージを含む zip ファイルをダウンロードしてください。

- (2) ダウンロードした zip ファイルをアップデート対象マシンに転送し、zip ファイルを展開します。

```
# unzip ScaTeFS_AI_Client_N.N.N.zip  
(Nは数字)
```

3.3.5 ScaTeFS クライアントのパッケージのインストール

以下のコマンドでインストールします。

- RHEL 8.6の場合

```
# dnf install scatefs-ai-client/el8.6/scatefs-ai-client/*.rpm
```

3.3.6 アップデート対象マシンの再起動

以下のコマンドで再起動します。

```
# reboot
```

3.3.7 ScaTeFS クライアントのセットアップ

「3.2 クライアントのセットアップ」を参照し、必要な設定を行ってください。

3.4 クライアントのアンインストール

3.4.1 ファイルシステムのアンマウントと scatefs-client サービスの停止

- (1) アンインストール対象マシンで ScaTeFS のファイルシステムを NFS でエクスポートしている場合、そのファイルシステムをマウントしている全ての NFS クライアントからそのファイルシステムをアンマウントしてください。アンマウントが完了したら (2) へ進んでください。

アンインストール対象マシンで ScaTeFS のファイルシステムを NFS でエクスポートしていない場合は、(3) に進んでください。

- (2) アンインストール対象マシンで nfs-server サービスを停止します。

```
# systemctl stop nfs-server
```

- (3) アンインストール対象マシンで ScaTeFS のファイルシステムを Samba で共有として公開している場合、その共有を使用している全ての Windows 等の CIFS クライアントでその共有へのアクセスを止めてください。アクセスの停止が完了したら (4) へ進んでください。

アンインストール対象マシンで ScaTeFS のファイルシステムを Samba で共有として公開していない場合は、(5) に進んでください。

- (4) アンインストール対象マシンで smb サービスと nmb サービスを停止します。

```
# systemctl stop smb
# systemctl stop nmb
```

- (5) アンインストール対象マシンで ScaTeFS のファイルシステムをアンマウントします。

```
# umount -a -t scatefs
```

- (6) アンインストール対象マシンで scatefs-client サービスを停止します。

```
# systemctl stop scatefs-client
```

3.4.2 ScaTeFS クライアントのパッケージのアンインストール

以下のコマンドでパッケージをアンインストールします。

```
# dnf remove '*scatefs-ai-client-*
```

第4章 ストレージ (V100) のセットアップと運用

本章では、IO サーバがアクセスするストレージ (iStorage V100) のセットアップ方法と運用方法について説明します。

4.1 ストレージ (V100) のセットアップ

4.1.1 IO ターゲットの設計

IO ターゲットは、ScaTeFS のファイルシステムの基盤となるデータストアです。クライアントノードから書き込まれたファイルデータは、IO サーバに分散され、さらに IO サーバ内の IO ターゲットに分散して格納されます。

IO ターゲットはファイルのデータ自体を格納するデータ領域と、ファイルタイプや更新時刻などを格納するメタデータ領域に分けられます。複数の IO ターゲットを作成できますが、必ずデータ領域とメタデータ領域の個数は同数で一对一の関係です。

4.1.2 に記載したストレージの設定は、データ領域およびメタデータ領域の IO ターゲット数をそれぞれ4にした場合の設定方法になります。

データ領域					メタデータ領域					IOターゲット数
SSD(NVMe)		パリティグループ			SSD(NVMe)		パリティグループ			
容量	ドライブ数	RAID	数	LDEV数	容量	ドライブ数	RAID	数	LDEV数	
1.9TB	16	RAID6 (14D+2P)	1	4	1.9TB	4	RAID1 (2D+2D)	1	1	4

IO ターゲット ID は、scatefs_addiot コマンド実行時に割り当てられます。以下に IO サーバが 2 台、IO ターゲットが 4 個のターゲット ID の割り当てになります。

IOサーバ	IOサーバ1	IOサーバ2
IOターゲットID	0	2
	1	3

4.1.2 ストレージの設定

iStorageV100 でボリュームを作成する場合、SVP 上の StorageNavigator を用いてボリュームを作成する必要があります。以下にデータ領域およびメタデータ領域の作成方法を記載します。

なお、詳細については iStorage V シリーズのマニュアル「システム管理者ガイド」、「HA Device Manager - Storage Navigator ユーザガイド」を参照してください。

4.1.2.1 パリティグループの作成

iStorage V シリーズでは、パリティグループ (PG) と呼ぶ同じ容量を持ち、1 つのデータグループとして扱われる一連のドライブで管理します。

パリティグループ作成の際、RAID レベルとドライブ選択が必要となります。パリティグループは以下の値を選択してください。

パラメータ	データ領域	メタデータ領域
RAIDレベル	RAID-6	RAID-1
ドライブ数 (構成)	16 (14D+2P)	4 (2D+2D)

4.1.2.2 プール LDEV の作成

プールを作成の際は、複数のプール LDEV を作成する (切り出す) 必要がある。プール LDEV 作成は以下の値で作成します。

パラメータ	データ領域	メタデータ領域
ドライブタイプ/インタフェース/RPM/容量	SSD/NVMe/-/1.9TB	SSD/NVMe/-/1.9TB
RAIDレベル	RAID-6	RAID-1
容量 (1 LDEV)	3000GB	800GB
LDEV数	8	4
LDEV名 (任意の名前)	SCATEFS-DATA1 SCATEFS-DATA2 SCATEFS-DATA3 SCATEFS-DATA4 SCATEFS-DATA5 SCATEFS-DATA6 SCATEFS-DATA7 SCATEFS-DATA8	SCATEFS-META1 SCATEFS-META2 SCATEFS-META3 SCATEFS-META4

4.1.2.3 プールの作成

プール LDEV を束ねてプールを作成します。プール作成は以下の値で作成します。

パラメータ	データ領域	メタデータ領域
ドライブタイプ/インタフェース/RPM	SSD/NVMe/-	SSD/NVMe/-
RAIDレベル	RAID-6	RAID-1
LDEV名 (4.1.2.2 で 設 定 し た LDEV名)	SCATEFS-DATA1 SCATEFS-DATA2 SCATEFS-DATA3 SCATEFS-DATA4	SCATEFS-META1 SCATEFS-META2 SCATEFS-META3 SCATEFS-META4
プール名 (任意の名前)	SCATEFS-DATA-POOL	SCATEFS-META-POOL
プールタイプ	Dynamic Provisioning	Dynamic Provisioning
複数階層プール	無効	無効
データダイレクトマップ	無効	無効

4.1.2.4 仮想ボリュームの作成

プールから仮想ボリュームを作成 (切り出す) します。仮想 LDEV は以下の値で作成します。

パラメータ	データ領域	メタデータ領域
プール名 (4.1.2.3で設定したプール名)	SCATEFS-DATA-POOL	SCATEFS-META-POOL
仮想LDEV容量 (1 仮想LDEV)	5000GB	600GB
LDEV数	4	4
LDEV名 (任意の名前)	SCATEFS-DATA-LDEV1 SCATEFS-DATA-LDEV2 SCATEFS-DATA-LDEV3 SCATEFS-DATA-LDEV4	SCATEFS-META-LDEV1 SCATEFS-META-LDEV2 SCATEFS-META-LDEV3 SCATEFS-META-LDEV4

4.1.2.5 ホストグループ作成

IO サーバからアクセスできるようにホストグループを設定します。WWPN 毎にホストグループを作成してください。設定値は以下の通りです。

パラメータ	IOサーバ1	IOサーバ2
ホストグループ名 (任意)	IOSV01-0 IOSV01-1	IOSV02-0 IOSV02-1
ホストモード	00 [Standard]	00 [Standard]
ホストモードオプション	2, 22, 25, 68	2, 22, 25, 68
利用可能なホスト	IOサーバ1のHBAの WWPN1 IOサーバ1のHBAの WWPN2	IOサーバ2のHBAの WWPN1 IOサーバ2のHBAの WWPN2
ポート : CL1-A (IOSV01-WWPN1) ポート : CL3-A (IOSV01-WWPN2) ポート : CL2-A (IOSV02-WWPN1) ポート : CL4-A (IOSV02-WWPN2)	セキュリティ : 有効 T10PIモード : 無効	セキュリティ : 有効 T10PIモード : 無効

4.1.2.6 LUN パスの追加

作成した仮想 LDEV に対して LUN パスを設定します。設定値は以下の通りです。

パラメータ	IOサーバ1	IOサーバ2
ホストグループ名 (4.1.2.5で設定したホストグループ名)	IOSV01-0 IOSV01-1	IOSV02-0 IOSV02-1
ポート	CL1-A,CL3-A	CL2-A,CL3-A
LDEV名 (4.1.2.4で設定した仮想LDEV名)	SCATEFS-DATA-LDEV1 SCATEFS-DATA-LDEV2 SCATEFS-DATA-LDEV3 SCATEFS-DATA-LDEV4 SCATEFS-META-LDEV1 SCATEFS-META-LDEV2 SCATEFS-META-LDEV3 SCATEFS-META-LDEV4	SCATEFS-DATA-LDEV1 SCATEFS-DATA-LDEV2 SCATEFS-DATA-LDEV3 SCATEFS-DATA-LDEV4 SCATEFS-META-LDEV1 SCATEFS-META-LDEV2 SCATEFS-META-LDEV3 SCATEFS-META-LDEV4

4.1.2.7 スペアドライブの割り当て

スペアドライブの割り当てを行います。StrageNavigator のパリティグループ作成から、未使用のドライブを選択し、スペアドライブの割り当てを行います。

4.1.3 LVM の設定

OS のディスク認識順序が変わっても LVM から見えるディスク順が変わらないようにするため、ディスクに対して一意で認識するようにします。

以下のコマンドを投入し、SCSI 論理デバイスに対して udev (by-id 名) を取得します。

例 SCSI デバイスの論理デバイスが/dev/sda2 の場合

```
# udevadm info -q symlink -n /dev/sda2
block/8:0
disk/by-id/scsi-3600605b005d7a320196f1f53484dfb20
disk/by-path/<ディスクの位置情報から生成される情報>
disk/by-uuid/<ディスクの固有情報>
```

```
disk/by-id/wwn-<ディスクの固有情報>
```

出力された 3600605b005d7a320196f1f53484dfb20 が, sda2 と対応する udev です。

本 udev 名を lvm へ認識させるため、lvm の設定 (global_filter) に udev を追加し、SCSI 論理デバイスを除外します。

合わせて、types、md_component_detection、allow_changes_with_duplicate_pvs の設定値を変更します。

lvm.conf

```
devices {
:
global_filter = [ ※
"a|sddlm\[a-p\]\[a-p\].\*|",
"a|scsi-3600605b005d7a320196f1f53484dfb20|",    /* by-id名を追加する */
:
"r|/dev/sda|",                                /* by-id名に対するSCSI論理デバイスを除外 */
:
]
types = [ "sddlmfdrv", 16 ]
md\_component\_detection=0
allow\_changes\_with\_duplicate\_pvs = 1
}
```

※global_filter の設定方法は以下のマニュアルを参照してください。

HA Command Suite Dynamic Link Manager ユーザーズガイド (Linux 用) 3.10.1. LVM2 を使用する場合の注意事項

付録 A CLUSTERPRO のクラスタ構成情報作成手順 (オフラインバージョン)

本手順書では、CLUSTERPRO X Cluster WebUI Offline を使用し、IO サーバ構築前にクラスタ構成情報を作成する手順について記載します。

以下の順序で CLUSTERPRO のクラスタ構成情報を作成していきます。

1. CLUSTERPRO ツールのインストール
2. CLUSTERPRO ツールの起動
3. クラスタ構成情報作成
 - 3.1 クラスタの作成
 - 3.2 フェイルオーバーグループの作成
 - 3.3 モニタリソースの作成
 - 3.4 モニタリソース異常時の回復動作設定
 - 3.5 クラスタプロパティの変更

本手順書は CLUSTERPRO のマニュアル「CLUSTERPRO X for Linux インストール&設定ガイド」を補完する位置づけですので、適宜同マニュアルを参照ください。本手順書では以下を参照ポイントとしています。

マニュアル (1) : 第 6 章クラスタ構成情報を作成する の 2 ノードクラスタ構成情報の作成手順

マニュアル (2) : 第 6 章クラスタ構成情報を作成する の クラスタ構成情報を保存する

また、本手順書で使用している CLUSTERPRO X Cluster WebUI Offline は以下となります。

CLUSTERPRO のサイトから入手します。

5.1.1-230601-1

A.1 はじめに

作業を始める前に事前に決めておく情報があります。これらの情報を決めた上で作業を始めてください。

- IO サーバ名
- クラスタを構成する 2 台の IO サーバ間のインタコネクト用 IP アドレス
- ファイルシステムポートのフローティング IP アドレス (FIP)
- 各種リソース名

これら各種リソースとリソース間の対応表を別紙に添付していますのでご参照ください。本表

で記載している各種リソース名は、推奨する命名規則で例示しています。特段の理由がない限り、本命名規則でリソース名を決めてください。

なお、ハートビート領域用のパーティションのデバイス名も決める必要がありますが、本名称は HDLM インストール後に決定されるため、「2.4.7 CLUSTERPRO」で設定します。

A.2 CLUSTERPROツールのインストール

CLUSTERPRO サイトの手順書「Cluster WebUI Offline 利用ガイド」を参照してください。

A.3 CLUSTERPROツールの起動

CLUSTERPRO サイトの手順書「Cluster WebUI Offline 利用ガイド」を参照してください。

A.4 クラスタ構成情報作成

「クラスタ生成ウィザード」を使用してクラスタ構成情報を作成します。

マニュアル（1）を参照してください。

以降の手順では、未記載の項目はデフォルトのままとしてください。

A.5 クラスタ

「クラスタ生成ウィザード」をクリックして、ウィザードを開始します。クラスタ画面はデフォルトのままで、[次へ] をクリックします。

A.6 サーバの基本設定

サーバの[基本設定]画面では、クラスタを構成する 2 台の IO サーバを追加します。以下の説明では IO サーバを iosv00,iosv01 と記載します。

[サーバの定義一覧]で [追加] をクリックします。

[サーバ追加]画面で以下の項目を設定します。マスタサーバとなります。

サーバ名

iosv00

再度、[サーバの定義一覧]で [追加] をクリックします。

[サーバ追加]画面で以下の項目を設定します。

サーバ名

iosv01

[次へ]をクリックします。

A.7 サーバのインタコネクトの設定

クラスタを構成する IO サーバ間のインタコネクトを設定します。

[インタコネクト一覧]で [追加] をクリックします。

優先度 1 の行の項目を設定します。

種別

カーネルモード

iosv00

IO サーバ間インタコネクトの IP アドレス

iosv01

IO サーバ間インタコネクトの IP アドレス

[インタコネクト一覧]で [追加] をクリックします。

優先度 2 の行の項目を設定します。

種別

カーネルモード

iosv00

運用・管理用ネットワークの IP アドレス

iosv01

運用・管理用ネットワークの IP アドレス

[インタコネクト一覧]で [追加] をクリックします。

優先度 3 の行の項目を設定します。

種別

DISK

iosv00

CLUSTERPRO のハートビート領域用のパーティションのデバイス名

iosv01

CLUSTERPRO のハートビート領域用のパーティションのデバイス名

※ハートビート領域用のパーティションのデバイス名は、2.4.8 で設定します。

A.8 サーバのフェンシングの設定

サーバの[フェンシング]画面はデフォルトのままで、[次へ] をクリックします。

A.9 フェイルオーバーグループの作成

クラスタを構成する IO サーバで動作するフェイルオーバーグループを作成します。 以下の説

明では iosv00 で動作するフェイルオーバーグループを failover1 、iosv01 で動作するフェイルオーバーグループを failover2 と記載します。

最初に iosv00 で動作するフェイルオーバーグループ failover1 を作成しますので、A.10 から A.14 を実施してください。A.10 から A.14 が完了しましたら、iosv01 で動作するフェイルオーバーグループ failover2 を作成しますので、再度 A.10 から A.14 を実施してください。

なお、フェイルオーバーグループ failover1 と フェイルオーバーグループ failover2 で異なる設定を行う項目については、各章の中で [failover1]、[failover2]と分けて明記しています。

A.10 フェイルオーバーグループの追加

[グループ一覧]で [追加] をクリックします。

[グループの定義]画面の[基本設定]で以下の項目を設定します。

名前

[failover1]

failover1

[failover2]

failover2

[起動可能サーバ] で [全てのサーバでフェイルオーバー可能] のチェックを外します。

[起動可能なサーバ] で 以下の順序で IO サーバを選択して [追加] をクリックします。

[failover1]

iosv00

iosv01

[failover2]

iosv01

iosv00

[グループ属性]で以下の項目をデフォルトから変更します。

フェイルバック属性

自動フェイルバック

A.11 グループリソース (フローティング IP リソース) の追加

IO サーバのネットワーク設定のファイルシステム用ネットワークの IP アドレスを設定します。なお、IO サーバのネットワーク設定のファイルシステムのポート数で追加するリソース数が異なりますが、本手順書では、10GbE の4つの FIP(fip1, fip2, fip3, fip4)について、[failover1]

へ fip1, fip2、[failover2]へ fip3, fip4 を追加する例を記載します。

[グループの定義]で [追加] をクリックします。

[グループのリソース定義]画面の [情報]で以下の項目を設定します。

タイプ

フローティング IP リソース

名前

[failover1]

fip1

fip2

[failover2]

fip3

fip4

[依存関係]画面はデフォルトのままで、[次へ] をクリックします。

[復旧動作]画面はデフォルトのままで、[次へ] をクリックします。

[詳細]画面 (共通タブ) で以下の項目を設定します。

IP アドレス

ファイルシステム用ネットワークの IP アドレス

設定例

10.0.1.1/25%ens2f0

bonding インターフェースの場合、[調整]ボタンをクリックします。

[フローティング IP リソース調整プロパティ]のパラメータタブで以下の項目を設定します。

NIC Link Down を異常と判定する

チェックボックスをオンにします。

A.12 グループリソース (ボリュームマネージャリソース) の追加

2.1.2 で設計した IO ターゲットのデータ/メタデータ領域で使用する VG について、グループリソースとして追加します。IO ターゲット数分のグループリソース (データ/メタデータ領域) を追加します。

[グループの定義]で [追加] をクリックします。

[グループのリソース定義]画面の [情報]で以下の項目を設定します。

タイプ

ボリュームマネージャリソース

名前

[failover1]

volmgr_d_01,volmgr_d_02,・・・,volmgr_d_[n]

volmgr_c_01,volmgr_c_02,・・・,volmgr_c_[n]

[failover2]

volmgr_d_[n+1],volmgr_d_[n+2],・・・,volmgr_d_[n+n]

volmgr_c_[n+1],volmgr_c_[n+2],・・・,volmgr_c_[n+n]

※[n]は IO サーバごとの IO ターゲット数です。

[依存関係]画面で以下の項目を設定します。

既定の依存関係に従う

チェックを外します。

[依存するリソース] で 利用可能なリソースのフローティング IP アドレスリソースを選択して [追加] をクリックします。

[failover1]

fip1, fip2

[failover2]

fip3, fip4

[復旧動作]画面はデフォルトのままで、[次へ] をクリックします。

[詳細]画面(共通タブ)で以下の項目を設定します。

ターゲット名

[failover1]

vg_data01,vg_data02,・・・,vg_data[n]

vg_ctrl01,vg_ctrl02,・・・,vg_ctrl[n]

[failover2]

vg_data[n+1],vg_data[n+2],・・・,vg_data[n+n]

vg_ctrl[n+1],vg_ctrl[n+2],・・・,vg_ctrl[n+n]

A.13 グループリソース (ディスクリソース) の追加

IO ターゲットのデバイスのマウント/アンマウントを行うリソースを追加します。IO ターゲット数分のグループリソース (データ/メタデータ領域) を作成します。

[グループの定義]で [追加] をクリックします。

[グループのリソース定義]画面の [情報]で以下の項目を設定します。

タイプ

ディスクリソース

名前

[failover1]

disk_d_01,disk_d_02, . . . ,disk_d_[n]

disk_c_01,disk_c_02, . . . ,disk_c_[n]

[failover2]

disk_d_[n+1],disk_d_[n+2], . . . ,disk_d_[n+n]

disk_c_[n+1],disk_c_[n+2], . . . ,disk_c_[n+n]

※[n]は IO サーバごとの IO ターゲット数です。

[依存関係]画面で以下の項目を設定します。

既定の依存関係に従う

チェックを外します。

[依存するリソース] で 利用可能なリソースのフローティング IP アドレスリソースを選択して [追加] をクリックします。

[failover1]

fip1, fip2, 対象のボリュームマネージャリソース

(たとえば disk_d_01 に対しては volmgr_d_01)

[failover2]

fip3, fip4, 対象のボリュームマネージャリソース

(たとえば disk_d_05 に対しては volmgr_d_05)

[復旧動作]画面はデフォルトのままで、[次へ] をクリックします。

[詳細]画面(共通タブ)で以下の項目を設定します。

ディスクのタイプ

lvm

ファイルシステム

データ領域

xfs

メタデータ領域

ext4

デバイス名

LV Path

※VG 名と LV 名からデバイス名を設定可能です (/dev/VG 名/LV 名)。

[failover1]

データ領域

/dev/vg_data01/lv_data01, . . . ,/dev/vg_data[n]/lv_data[n]

メタデータ領域

/dev/vg_ctrl01/lv_ctrl01, . . . ,/dev/vg_ctrl[n]/lv_ctrl[n]

[failover2]

データ領域

/dev/vg_data[n+1]/lv_data[n+1], . . . , /dev/vg_data[n+n]/lv_data[n+n]

メタデータ領域

/dev/vg_ctrl[n+1]/lv_ctrl[n+1], . . . , /dev/vg_ctrl[n+n]/lv_ctrl[n+n]

マウントポイント

データ領域

/mnt/iot/X/data

メタデータ領域

/mnt/iot/X/ctrl

※X は IO ターゲット ID を指定します。

ファイルシステムへ ext4 を指定した場合、[詳細]画面(共通タブ)の[調整]ボタンをクリックします。

[ディスクリソース調整プロパティ]の Fsck タブで以下の項目を設定します。

Mount 実行前の fsck アクション

実行しない

A.14 グループリソース (EXEC リソース) の追加

IO サーバで実行するリソース (ルーティング、ScaTeFS サーバ) を追加します。

複数のサブネット環境の場合は、ルーティングの EXEC リソースを追加します。

- ルーティング

[グループの定義]で [追加] をクリックします。

[グループのリソース定義]画面の [情報]で以下の項目を設定します。

タイプ

EXEC リソース

名前

[failover1]

exec_route1

[failover2]

exec_route2

[依存関係]画面で以下の項目を設定します。

既定の依存関係に従う

チェックを外します。

[依存するリソース] で 利用可能なリソースのフローティング IP アドレスリソースを選択して [追加] をクリックします。

[failover1]

fip1, fip2

[failover2]

fip3, fip4

[復旧動作]画面はデフォルトのままで、[次へ] をクリックします。

[詳細]画面（共通タブ）で以下の項目を設定します。

ユーザアプリケーションを選択します。

[スクリプト一覧]で[編集]をクリックします。

[アプリケーションパスの入力]で以下の項目を設定します。

開始

/opt/scatefs/script/exec_route.sh

- ScaTeFSサーバ

[グループの定義]で [追加] をクリックします。

[グループのリソース定義]画面の [情報]で以下の項目を設定します。

タイプ

EXEC リソース

名前

[failover1]

exec1

[failover2]

exec2

[依存関係]画面で以下の項目を設定します。

既定の依存関係に従う

チェックを外します。

[依存するリソース] で 利用可能なリソースのフローティング IP アドレスリソースを選択して [追加] をクリックします。

※表示されるすべてのリソースを追加します。

[failover1]

fip1, fip2, ボリュームマネージャリソース, ディスクリソース, EXEC リソース (ルーティング)

[failover2]

fip3, fip4, ボリュームマネージャリソース, ディスクリソース, EXEC リソース (ルー

ディング)

[復旧動作]画面はデフォルトのままで、[次へ] をクリックします。

[詳細]画面(共通タブ)で以下の項目を設定します。

ユーザアプリケーションを選択します。

[スクリプト一覧]で[編集]をクリックします。

[アプリケーションパスの入力]で以下の項目を設定します。

開始

/opt/scatefs/script/start.sh

終了

/opt/scatefs/script/stop.sh

A.15 モニタリソース (ディスクモニタ) 追加

ディスクリソースのモニタリソースを追加します。

IO サーバごとにメタデータ領域のディスクリソースの最初の 1 つを追加します。

iosv00

disk_c_01

iosv01

disk_c_[n+1]

※[n]は IO サーバごとの IO ターゲット数です。

[モニタ]画面の[モニタリソース一覧]で [追加] をクリックします。

[モニタリソースの定義]画面で以下の項目を設定します。

タイプ

ディスク RW 監視

名前

iosv00

diskw_c_01

iosv01

diskw_c_[n+1]

[監視 (共通)]画面で以下の項目を設定します。

監視タイミグ

活性時を選択します。

対象リソース

iosv00

[参照]をクリックして、disk_c_01 を選択します。

iosv01

[参照]をクリックして、disk_c_[n+1]を選択します。

[監視 (固有)]画面の共通タブで以下の項目を設定します。

監視先

iosv00

/dev/vg_ctrl01/lv_ctrl01

iosv01

/dev/vg_ctrl[n+1]/lv_ctrl[n+1]

[回復動作]画面で以下の項目を設定します。

回復対象

iosv00

[参照]をクリックして、disk_c_01 を選択します。

iosv01

[参照]をクリックして、disk_c_[n+1]を選択します。

A.16 モニタリソース (カスタムモニタ) 追加

EXEC リソース (ScaTeFS サーバ) のモニタリソースを追加します。

[モニタ]画面の[モニタリソース一覧]で [追加] をクリックします。

[モニタリソースの定義]画面で以下の項目を設定します。

タイプ

カスタム監視

名前

iosv00

genw1

iosv01

genw2

[監視 (共通)]画面で以下の項目を設定します。

インターバル

15

監視タイミング

活性時を選択します。

対象リソース

iosv00

[参照]をクリックして、exec1 を選択します。

iosv01

[参照]をクリックして、exec2 を選択します。

[監視 (固有)]画面の共通タブで以下の項目を設定します。

ユーザアプリケーションを選択します。

ファイル

iosv00

/opt/scatefs/script/is_exec1_ios_running.sh

iosv01

/opt/scatefs/script/is_exec2_ios_running.sh

[回復動作]画面で以下の項目を設定します。

回復動作

回復対象を再起動、効果がなければフェイルオーバー実行

回復対象

iosv00

[参照]をクリックして、exec1 を選択します。

iosv01

[参照]をクリックして、exec2 を選択します。

A.17 モニタリソース (ボリュームマネージャモニタ) 設定変更

自動で作成されたすべてのボリュームマネージャモニタの設定を変更します。

[モニタ]画面の[モニタリソース一覧]で volmgrwX を選択して、[プロパティ]をクリックします。

[監視 (共通)]画面で以下の項目を設定します。

タイムアウト

240

リトライ回数

3

A.18 モニタリソース (ユーザ空間モニタ) 設定変更

自動で作成されたユーザ空間モニタの設定を変更します。

[モニタ]画面の[モニタリソース一覧]で userw を選択して、[プロパティ]をクリックします。

[監視 (固有)]画面で以下の項目を設定します。

タイムアウト発生時動作

PANIC

監視の拡張設定

以下の項目について、チェックします。

ダミーファイルのオープン/クローズ

書き込みを行う

A.19 モニタリソース (フローティングIPモニタ) 設定変更

bonding インターフェースの場合、フローティング IP モニタの設定を変更します。

[モニタ]画面の[モニタリソース一覧]で fipw1~fipwX をそれぞれ選択して、[プロパティ]をクリックします。

[監視 (固有)]画面で以下の項目を設定します。

NIC Link Up/Down を監視する

チェックボックスをオンにします。

A.20 モニタリソース (IPモニタ)の追加

複数のサブネット環境の場合は、フローティング IP リソースごとに IP モニタリソースを追加します。

[モニタ]画面の[モニタリソース一覧]で [追加] をクリックします。

[モニタリソースの定義]画面で以下の項目を設定します。

タイプ

IP 監視

名前

iosv00

ipw1, ipw2

iosv01

ipw3, ipw4

[監視 (共通)]画面で以下の項目を設定します。

インターバル

30

リトライ回数

3

監視タイミング

活性時を選択します。

対象リソース

iosv00

ipw1 : [参照]をクリックして、fip1 を選択します。

ipw2 : [参照]をクリックして、fip2 を選択します。

iosv01

ipw3 : [参照]をクリックして、fip3 を選択します。

ipw4 : [参照]をクリックして、fip4 を選択します。

監視を行うサーバを選択するの[サーバ]をクリックします。

[異常検出サーバ]画面で[独自に設定する]

[起動可能なサーバ] で 以下の IO サーバを選択して [追加] をクリックします。

ipw1, ipw2

iosv00

ipw3, ipw4

iosv01

[監視 (固有)]画面の共通タブで以下の項目を設定します。

[追加]をクリックして IP アドレスに以下のように入力します。

ipw1

fip1 が接続されるネットワークのゲートウェイ

ipw2

fip2 が接続されるネットワークのゲートウェイ

ipw3

fip3 が接続されるネットワークのゲートウェイ

ipw4

fip4 が接続されるネットワークのゲートウェイ

[回復動作]画面で以下の項目を設定します。

回復動作

回復対象を再起動、効果がなければフェイルオーバー実行

回復対象

ipw1

[参照]をクリックして、fip1 を選択します。

ipw2

[参照]をクリックして、fip2 を選択します。

ipw3

[参照]をクリックして、fip3 を選択します。

ipw4

[参照]をクリックして、fip4 を選択します。

A.21 モニタリソース異常時の回復動作設定

モニタリソースを作成し、[完了] をクリックすると、以下のポップアップメッセージが表示されますので、[はい] をクリックします。

下記の動作を有効にしますか？

- ・グループの自動起動
- ・グループリソース活性・非活性異常時の復旧動作
- ・モニタリソース異常時の回復動作

A.22 クラスタプロパティの変更

クラスタ (cluster) のプロパティを開きます。

タイムアウトタブを選択して以下の項目を設定します。

内部通信タイムアウト

300

以上で作業は完了となりますが、作成したクラスタ構成情報をファイルシステムへ保存してください。この後の IO サーバ構築で、クラスタ構成情報ファイルをインポートする際に使用します。

「設定のエクスポート」をクリックして、任意のディレクトリに保存してください。

マニュアル（2）を参照してください。

A.23 クラスタ構成情報のパラメーター一覧

本手順書で作成するクラスタ構成情報のパラメータを記載します。リソース名や LVM 名の設定値を推奨します。また、IP アドレスの設定値は例として記載しますので、ネットワーク設計の値へ読み替えてください。

- クラスタのプロパティ

[インタコネクトタブ]

優先度	種別	iosv00	iosv01
1	カーネルモード	IOサーバ間インタコネクトのIPアドレス	IOサーバ間インタコネクトのIPアドレス
2	カーネルモード	運用・管理用ネットワークのIPアドレス	運用・管理用ネットワークのIPアドレス
3	DISK	ハートビート領域用のパーティションのデバイス名	ハートビート領域用のパーティションのデバイス名

[タイムアウトタブ]

項目	設定値
内部通信タイムアウト	300

- サーバのプロパティ

[マスタサーバタブ]

順位	名前
マスタサーバ	iosv00
1	iosv01

- グループのプロパティ

画面	項目	設定値	
		iosv00	iosv01
グループの定義	名前	failover1	failover2
起動可能サーバ	全てのサーバでフェイルオーバー可能	チェックを外します	
	順位1	iosv00	iosv01
	順位2	iosv01	iosv00
グループ属性	フェイルバック属性	自動フェイルバック	

● グループリソース : フローティング IP リソース

画面	項目	設定値	
		failover1	failover2
グループリソースの定義	名前	fip1, fip2	fip3, fip4
詳細	IPアドレス	10.0.1.1/25%ens2f0 10.0.2.1/25%ens2f1	10.0.1.2/25%ens2f0 10.0.2.2/25%ens2f1

● グループリソース : ボリュームマネージャリソース

[データ領域]

画面	項目	設定値	
		failover1	failover2
グループリソースの定義	名前	volmgr_d_01 volmgr_d_02 ... volmgr_d_[n]	volmgr_d_[n+1] volmgr_d_[n+2] ... volmgr_d_[n+n]
依存関係	依存するリソース	fip1, fip2	fip3, fip4
詳細	ターゲット名	vg_data01 vg_data02 ... vg_data[n]	vg_data[n+1] vg_data[n+2] ... vg_data[n+n]

[メタデータ領域]

画面	項目	設定値	
		failover1	failover2
グループリソースの定義	名前	volmgr_c_01 volmgr_c_02 ... volmgr_c_[n]	volmgr_c_[n+1] volmgr_c_[n+2] ... volmgr_c_[n+n]
依存関係	依存するリソース	fip1, fip2	fip3, fip4
詳細	ターゲット名	vg_ctrl01 vg_ctrl02 ... vg_ctrl[n]	vg_ctrl[n+1] vg_ctrl[n+2] ... vg_ctrl[n+n]

● グループリソース : ディスクリソース

[データ領域]

画面	項目	設定値	
		failover1	failover2
グループリソースの定義	名前	disk_d_01 disk_d_02 ... disk_d_[n]	disk_d_[n+1] disk_d_[n+2] ... disk_d_[n+n]
依存関係	依存するリソース	fip1, fip2 対象のボリュームマネージャリソース (volmgr_d_XX)	fip3, fip4 対象のボリュームマネージャリソース (volmgr_d_XX)
詳細	ディスクタイプ	lvm	
	ファイルシステム	xfs	
	デバイス名	/dev/vg_data01/lv_data01 /dev/vg_data02/lv_data02 ... /dev/vg_data[n]/lv_data[n]	/dev/vg_data[n+1]/lv_data[n+1] /dev/vg_data[n+2]/lv_data[n+2] ... /dev/vg_data[n+n]/lv_data[n+n]
	マウントポイント	/mnt/iot/0/data /mnt/iot/1/data ... /mnt/iot/[n-1]/data	/mnt/iot/[n]/data /mnt/iot/[n+1]/data ... /mnt/iot/[n+n-1]/data

[メタデータ領域]

画面	項目	設定値	
		failover1	failover2
グループリソースの定義	名前	disk_c_01 disk_c_02 ... disk_c_[n]	disk_c_[n+1] disk_c_[n+2] ... disk_c_[n+n]
依存関係	依存するリソース	fip1, fip2 対象のボリュームマネージャリソース (volmgr_c_XX)	fip3, fip4 対象のボリュームマネージャリソース (volmgr_c_XX)
詳細	ディスクタイプ	lvm	
	ファイルシステム	ext4	

画面	項目	設定値	
		failover1	failover2
	デバイス名	/dev/vg_ctrl01/lv_ctrl01 /dev/vg_ctrl02/lv_ctrl02 ... /dev/vg_ctrl[n]/lv_ctrl[n]	/dev/vg_ctrl[n+1]/lv_ctrl[n+1] /dev/vg_ctrl[n+2]/lv_ctrl[n+2] ... /dev/vg_ctrl[n+n]/lv_ctrl[n+n]
	マウント ポイント	/mnt/iot/0/ctrl /mnt/iot/1/ctrl ... /mnt/iot/[n-1]/ctrl	/mnt/iot/[n]/ctrl /mnt/iot/[n+1]/ctrl ... /mnt/iot/[n+n-1]/ctrl

マウントポイントパス中の IO ターゲット ID(/mnt/iot/X/)について、IO ターゲット作成時にシステムで割り当てる IO ターゲット ID は、先頭の IO サーバ(iosv00)から順に 0 から割り当てます。

以下に IO サーバに IO ターゲットを n 個構築した場合の IO ターゲット ID の割り当てを記載します。

iosv00

IO ターゲット ID : 0 ~ n-1

iosv01

IO ターゲット ID : 0 ~ n-1

意図どおりに IO ターゲット ID が割り振られているか、IO ターゲット作成後、scatefs_detail -t コマンドを使用して確認してください。コマンドの使用方法は、「2.4.5 IO ターゲットの登録」を参照してください。

- グループリソース : EXEC リソース
[ルーティング]

画面	項目	設定値	
		failover1	failover2
グループリソースの 定義	名前	exec_route1	exec_route2
依存関係	依存する リソース	fip1, fip2	fip3, fip4
詳細	スクリプト	ユーザアプリケーション	
アプリケーションパスの入力	開始	/opt/scatefs/script/exec_route.sh	

[ScaTeFS サーバ]

画面	項目	設定値	
		failover1	failover2
グループリソースの定義	名前	exec1	exec2
依存関係	依存するリソース	すべて	すべて
詳細	スクリプト	ユーザアプリケーション	
アプリケーションパスの入力	開始	/opt/scatefs/script/start.sh	
	終了	/opt/scatefs/script/stop.sh	

● モニタリソース : ディスクモニタ

画面	項目	設定値	
		failover1	failover2
モニタリソースの定義	名前	diskw_c_01	diskw_c_[n+1]
監視(共通)	監視タイミング	活性時	
	対象リソース	disk_c_01	disk_c_[n+1]
監視(固有)	監視先	/dev/vg_ctrl01/lv_ctrl01	/dev/vg_ctrl[n+1]/lv_ctrl[n+1]
回復動作	回復対象	disk_c_01	disk_c_[n+1]

● モニタリソース : カスタムモニタ

画面	項目	設定値	
		failover1	failover2
モニタリソースの定義	名前	genw1	genw2
監視 (共通)	インターバル	15	
	監視タイミング	活性時	
	対象リソース	exec1	exec2
監視 (固有)	スクリプト	ユーザアプリケーション	
	ファイル	/opt/scatefs/script/is_exec1_i os_running.sh	/opt/scatefs/script/is_exec2_i os_running.sh
	監視タイミング	同期	
回復動作	回復動作	回復対象を再起動、効果がなければフェイルオーバー実行	
	回復対象	exec1	exec2

● モニタリソース : ボリュームマネージャモニタ

画面	項目	設定値
監視 (共通)	タイムアウト	240
	リトライ回数	3

● モニタリソース : ユーザ空間モニタ

画面	項目	設定値
監視 (固有)	タイムアウト発生時動作	PANIC
	ダミーファイルのオープン/クローズ	チェックします
	書き込みを行う	チェックします

- モニタリソース：フローティングIPモニタ

bonding インターフェースの場合、フローティング IP モニタの設定を変更します。

画面	項目	設定値
監視 (固有)	NIC Link Up/Downを監視する	チェックします

- モニタリソース：IPモニタ

画面	項目	設定値	
		failover1	failover2
モニタリソースの定義	名前	ipw1, ipw2	ipw3, ipw4
監視 (共通)	インターバル	30	
	リトライ回数	3	
	監視タイミング	活性時	
	対象リソース	fip1, fip2	fip3, fip4
異常検出サーバ	起動可能なサーバ	iosv00	iosv01
監視 (固有)	IPアドレス	ネットワークのゲートウェイ	ネットワークのゲートウェイ
回復動作	回復動作	回復対象を再起動、効果がなければフェイルオーバー実行	
	回復対象	fip1, fip2	fip3, fip4

付録 B 発行履歴

B.1 発行履歴一覧表

2024 年 1 月 初版

B.2 追加・変更点詳細

- 初版
新規作成

NEC Scalable Technology File System for AI

(ScaTeFS for AI)

インストールガイド

2024年 1月 初版

日本電気株式会社

東京都港区芝五丁目 7 番 1 号

TEL(03)3454-1111 (大代表)

© NEC Corporation 2024

日本電気株式会社の許可なく複製・改変などを行うことはできません。

本書の内容に関しては将来予告なしに変更することがあります。