



ノードAPIメソッド Element Software

NetApp
November 12, 2025

目次

ノードAPIメソッド	1
CheckPingOnVlan	1
パラメータ	1
戻り値	2
リクエスト例	3
応答例	3
バージョン以降の新機能	4
提案ノード追加のチェック	4
パラメータ	5
戻り値	5
リクエスト例	6
応答例	7
バージョン以降の新機能	7
クラスターサポートバンドルの作成	7
パラメータ	7
戻り値	9
リクエスト例	9
応答例	9
バージョン以降の新機能	9
サポートバンドルの作成	10
パラメータ	10
戻り値	10
リクエスト例	11
応答例	11
バージョン以降の新機能	12
すべてのサポートバンドルを削除	12
パラメータ	12
戻り値	12
リクエスト例	12
応答例	13
バージョン以降の新機能	13
メンテナンスモードを無効にする	13
パラメータ	13
戻り値	13
リクエスト例	15
応答例	15
バージョン以降の新機能	16
詳細情報の参照	16
SSHを無効にする	16

パラメータ	16
戻り値	16
リクエスト例	16
応答例	17
メンテナンスモードを有効にする	17
パラメータ	17
戻り値	18
リクエスト例	19
応答例	20
バージョン以降の新機能	20
詳細情報の参照	20
SSHを有効にする	20
パラメータ	20
戻り値	21
リクエスト例	21
応答例	21
クラスタ構成の取得	21
パラメータ	21
戻り値	21
リクエスト例	22
応答例	22
バージョン以降の新機能	23
クラスター状態を取得する	23
パラメータ	23
戻り値	23
リクエスト例	23
応答例	23
バージョン以降の新機能	24
ゲットコンフィグ	24
パラメータ	24
戻り値	24
リクエスト例	24
応答例	25
バージョン以降の新機能	25
詳細情報の参照	25
ドライブ構成を取得	25
パラメータ	25
戻り値	25
リクエスト例	25
応答例	26
ハードウェア構成の取得	28

パラメータ	28
戻り値	28
リクエスト例	28
応答例	28
バージョン以降の新機能	30
ハードウェア情報を取得する	30
パラメータ	30
戻り値	30
リクエスト例	31
応答例	31
バージョン以降の新機能	32
GetIpmiConfig	32
パラメータ	32
戻り値	33
リクエスト例	33
応答例	33
バージョン以降の新機能	37
GetIpmiInfo	37
パラメータ	37
戻り値	37
リクエスト例	37
応答例	37
バージョン以降の新機能	40
ネットワーク構成の取得	40
パラメータ	41
戻り値	41
リクエスト例	41
応答例	41
バージョン以降の新機能	41
詳細情報の参照	41
ネットワークインターフェースの取得	41
パラメータ	41
戻り値	42
リクエスト例	42
応答例	43
バージョン以降の新機能	45
GetNodeActiveTlsCiphers	45
パラメータ	45
戻り値	45
リクエスト例	45
応答例	45

GetNodeFipsDrivesReport	46
パラメータ	46
戻り値	46
リクエスト例	47
応答例	47
バージョン以降の新機能	47
ノードSSL証明書の取得	48
パラメータ	48
戻り値	48
リクエスト例	48
応答例	48
GetNodeSupportedTlsCiphers	50
パラメータ	50
戻り値	50
リクエスト例	50
応答例	50
パッチ情報を取得	51
パラメータ	51
戻り値	52
リクエスト例	52
応答例	52
バージョン以降の新機能	53
保留中の操作を取得する	53
パラメータ	53
戻り値	53
リクエスト例	54
応答例	54
バージョン以降の新機能	54
GetSshInfo	54
パラメータ	55
戻り値	55
リクエスト例	55
応答例	55
ドライブハードウェアのリスト	55
パラメータ	56
戻り値	56
リクエスト例	56
応答例	56
バージョン以降の新機能	57
詳細情報の参照	58
ネットワークインターフェースの一覧	58

パラメータ	58
戻り値	58
リクエスト例	58
応答例	59
バージョン以降の新機能	60
ネットワークインターフェース統計の一覧	60
パラメータ	61
戻り値	61
リクエスト例	61
応答例	61
バージョン以降の新機能	62
リストテスト	62
パラメータ	62
戻り値	62
リクエスト例	63
応答例	63
バージョン以降の新機能	63
リストユーティリティ	63
パラメータ	63
戻り値	64
リクエスト例	64
応答例	64
バージョン以降の新機能	64
ノードSSL証明書の削除	65
パラメータ	65
戻り値	65
リクエスト例	65
応答例	65
リセットドライブ	65
パラメータ	66
戻り値	66
リクエスト例	66
応答例	66
バージョン以降の新機能	67
リセットノード	67
パラメータ	67
戻り値	68
リクエスト例	68
応答例	68
バージョン以降の新機能	70
リセットノード補足Tls暗号	70

パラメータ	70
戻り値	70
リクエスト例	70
応答例	70
再起動ネットワーク	71
パラメータ	71
戻り値	71
リクエスト例	71
応答例	71
バージョン以降の新機能	72
サービスの再起動	72
パラメータ	72
戻り値	72
リクエスト例	72
応答例	73
バージョン以降の新機能	73
クラスタ構成の設定	73
パラメータ	73
戻り値	74
リクエスト例	74
応答例	74
バージョン以降の新機能	75
設定	75
パラメータ	75
戻り値	76
リクエスト例	76
応答例	77
バージョン以降の新機能	77
詳細情報の参照	77
ネットワーク設定の設定	77
パラメータ	77
戻り値	78
リクエスト例	78
応答例	79
バージョン以降の新機能	79
詳細情報の参照	79
ノードSSL証明書の設定	79
パラメータ	79
戻り値	80
リクエスト例	80
応答例	81

ノード補足Tls暗号の設定	82
パラメータ	82
戻り値	82
リクエスト例	82
応答例	83
シャットダウン	83
パラメータ	84
戻り値	84
リクエスト例	84
応答例	84
バージョン以降の新機能	85
テストコネクトアンサンプル	85
パラメータ	85
戻り値	85
リクエスト例	86
応答例	86
バージョン以降の新機能	87
テストコネクトMvip	87
パラメータ	87
戻り値	87
リクエスト例	90
応答例	90
バージョン以降の新機能	91
テスト接続Svip	91
パラメータ	91
戻り値	92
リクエスト例	95
応答例	95
バージョン以降の新機能	96
テストドライブ	96
パラメータ	96
戻り値	97
リクエスト例	97
応答例	97
バージョン以降の新機能	97
テストハードウェア構成	97
パラメータ	97
戻り値	98
リクエスト例	98
応答例	99
バージョン以降の新機能	99

詳細情報の参照	99
テスト配置クラスター	99
パラメータ	99
戻り値	99
リクエスト例	99
応答例	99
バージョン以降の新機能	100
ローカル接続のテスト	100
パラメータ	100
戻り値	100
リクエスト例	101
応答例	101
バージョン以降の新機能	103
テストネットワーク構成	103
パラメータ	103
戻り値	103
リクエスト例	103
応答例1	104
応答例2	104
応答例3	105
バージョン以降の新機能	105
詳細情報の参照	105
テストピング	105
パラメータ	105
戻り値	107
リクエスト例	107
応答例	107
バージョン以降の新機能	108
リモート接続のテスト	108
パラメータ	109
戻り値	109
リクエスト例	109
応答例	109
バージョン以降の新機能	111

ノードAPIメソッド

CheckPingOnVlan

使用することができます `CheckPingOnVlan` 展開前のネットワーク検証を実行するときに、一時的な VLAN 上のネットワーク接続をテストする方法。`CheckPingOnVlan` 一時的な VLAN インターフェイスを作成し、VLAN インターフェイスを使用してストレージクラスター内のすべてのノードに ICMP パケットを送信してから、インターフェイスを削除します。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
試み	システムがテスト ping を繰り返す回数を指定します。	integer	5	いいえ
ホスト	ping するデバイスのアドレスまたはホスト名をコンマで区切ったリストを指定します。	string	クラスター内のノード	いいえ
インタフェース	ping を送信する既存の (ベース) インターフェイス。有効な値は次のとおりです。 <ul style="list-style-type: none">• Bond10G: Bond10G インターフェイスから ping を送信します。• Bond1G: Bond1G インターフェイスから ping を送信します。	string	なし	はい

Name	説明	タイプ	デフォルト値	必須
パケットサイズ	各 IP に送信される ICMP パケットで送信するバイト数を指定します。バイト数は、ネットワーク構成で指定された最大 MTU 未満である必要があります。	integer	なし	いいえ
pingタイムアウトミリ秒	個々の ping 応答を待機するミリ秒数を指定します。	integer	500ms	いいえ
断片化を禁止する	ICMP パケットの DF (Do not Fragment) フラグを有効にします。	ブーリアン	false	いいえ
ソースアドレスV4	ICMP ping パケットで使用する送信元 IPv4 アドレス。	string	なし	はい
ソースアドレスV6	ICMP ping パケットで使用する送信元 IPv6 アドレス。	string	なし	はい
合計タイムアウト秒	次の ping 試行を発行するかプロセスを終了する前に、ping がシステム応答を待機する時間を秒単位で指定します。	integer	5	いいえ
仮想ネットワークタグ	ping パケットを送信するときに使用する VLAN ID。	integer	なし	はい

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
結果	ノードが通信できた各 IP のリストと ping 応答統計。	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "CheckPingOnVlan",
  "params": {
    "interface": "Bond10G",
    "virtualNetworkTag": 4001,
    "sourceAddressV4": "192.168.41.4",
    "hosts": "192.168.41.2"
  },
  "id": 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```

{
  "id": 1,
  "result": {
    "192.168.41.2": {
      "individualResponseCodes": [
        "Success",
        "Success",
        "Success",
        "Success",
        "Success"
      ],
      "individualResponseTimes": [
        "00:00:00.000373",
        "00:00:00.000098",
        "00:00:00.000097",
        "00:00:00.000074",
        "00:00:00.000075"
      ],
      "individualStatus": [
        true,
        true,
        true,
        true,
        true
      ],
      "interface": "Bond10G",
      "responseTime": "00:00:00.000143",
      "sourceAddressV4": "192.168.41.4",
      "successful": true,
      "virtualNetworkTag": 4001
    }
  }
}

```

バージョン以降の新機能

11.1

提案ノード追加のチェック

使用することができます `CheckProposedNodeAdditions` ストレージ ノードのセットをテストして、エラーやベスト プラクティス違反なしでストレージ クラスターに追加できるかどうかを確認する方法。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
ノード	ストレージ クラスターに追加する準備ができているストレージ ノードのストレージ IP アドレスのリスト。	文字列配列	なし	はい

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
提案されたクラスター有効	提案されたストレージ ノードが有効なストレージ クラスターを構成するかどうかを示します。有効な値は次のとおりです。 <ul style="list-style-type: none">• true• false	ブーリアン

<p>提案されたクラスターエラー</p>	<p>提案されたストレージ ノードを使用してストレージ クラスターを作成した場合に発生するエラー。考えられるエラーコード:</p> <ul style="list-style-type: none"> • <code>nodesNoCapacity</code>: ノードには使用可能な容量がありませんでした。 • <code>nodesTooLarge</code>: アクティブ保護スキームでは、ノードがクラスター容量の大部分を占めすぎます。 • <code>nodesConnectFailed</code>: ハードウェア構成を照会するためにノードに接続できませんでした。 • <code>nodesQueryFailed</code>: ノードのハードウェア構成を照会できませんでした。 • <code>nodesClusterMember</code>: ノードの IP アドレスはクラスター内ですでに使用されています。 • <code>nonFipsNodeCapable</code>: FIPS 140-2 ドライブ暗号化機能が有効になっている間は、FIPS 非対応ノードをストレージ クラスターに追加できません。 • <code>nonFipsDrivesCapable</code>: FIPS 140-2 ドライブ暗号化機能が有効になっている間は、FIPS 非対応ドライブを搭載したノードをクラスターに追加できません。 	<p>文字列配列</p>
----------------------	--	--------------

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "CheckProposedNodeAdditions",
  "params": {
    "nodes": [
      "192.168.1.11",
      "192.168.1.12",
      "192.168.1.13",
      "192.168.1.14"
    ]
  },
  "id": 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "proposedClusterValid": true,
    "proposedClusterErrors": [ ]
  }
}
```

バージョン以降の新機能

11.0

クラスターサポートバンドルの作成

使用することができます `CreateClusterSupportBundle` 管理ノードで、クラスター内のすべてのノードからサポート バンドルを収集します。個々のノード サポート バンドルは、tar.gz ファイルとして圧縮されます。クラスター サポート バンドルは、ノード サポート バンドルを含む tar ファイルです。このメソッドは管理ノードでのみ実行できます。ストレージノードで実行した場合は機能しません。

パラメータ



このメソッドは管理ノードに対して呼び出す必要があります。例えば：

```
https://<management node IP>:442/json-rpc/10.0
```


このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
不完全な許可	1 つ以上のノードからバンドルを収集できない場合でも、スクリプトの実行を継続できます。	ブーリアン	なし	いいえ
バンドル名	作成された各サポート バンドルの一意の名前。名前が指定されていない場合は、「supportbundle」とノード名がファイル名として使用されます。	string	なし	いいえ
mvip	クラスターの MVIP。バンドルはクラスター内のすべてのノードから収集されます。 nodes パラメータが指定されていない場合は、このパラメータは必須です。	string	なし	はい
ノード	バンドルを収集するノードの IP アドレス。バンドルを収集するノードを指定するには、nodes または mvip のいずれかを使用します (両方は使用できません)。 mvip が指定されていない場合は、このパラメータは必須です。	文字列配列	なし	はい
パスワード	クラスター管理者のパスワード。 注意: このパスワードは入力するとテキストとして表示されます。	string	なし	はい
ユーザ名	クラスター管理者のユーザー名。	string	なし	はい

戻り値

このメソッドには戻り値はありません。

リクエスト例

このメソッドのリクエストは次の例ようになります。

```
{
  "method": "CreateClusterSupportBundle",
  "params": {
    "bundlename": "clusterbundle",
    "mvip": "132.119.120.100"
  },
  "id": 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "details": {
      "bundleName": "clusterbundle",
      "extraArgs": "",
      "files": [
        "/tmp/supportbundles/clusterbundle.cl-4SD5.tar"
      ],
      "output": "timeout -s KILL 1790s
/usr/local/bin/sfclustersupportbundle --quiet --name=\"clusterbundle\"
--target-directory=\"/tmp/solidfire-dtemp.MM7f0m\" --user=\"admin\"
--pass=\"admin\" --mvip=132.119.120.100"
    },
    "duration": "00:00:24.938127",
    "result": "Passed"
  }
}
```

バージョン以降の新機能

9.6

サポートバンドルの作成

使用できます `CreateSupportBundle` ノードのディレクトリの下にサポート バンドル ファイルを作成します。作成後、バンドルは tar ファイルとしてノードに保存されます (gz 圧縮オプションは extraArgs パラメータ経由で使用できます)。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
バンドル名	サポート バンドルの一意の名前。名前が指定されていない場合は、「supportbundle」とノード名がファイル名として使用されます。	string	なし	いいえ
追加引数	サポート バンドルを tar.gz ファイルとして作成するには、「--compress gz」を使用します。	string	なし	いいえ
タイムアウト秒	サポート バンドルスクリプトが実行される秒数。	integer	1500	いいえ

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
------	----	-----

詳細	<p>サポート バンドルの詳細。有効な値は次のとおりです。</p> <ul style="list-style-type: none"> • bundleName: CreateSupportBundleAPI メソッドで指定された名前。名前が指定されていない場合は、「supportbundle」が使用されます。 • extraArgs: このメソッドで渡される引数。 • files: システムによって作成されたサポート バンドル ファイルのリスト。 • 出力: サポート バンドルを作成したスクリプトからのコマンドライン出力。 • timeoutSec: サポート バンドル スクリプトが停止するまでの実行秒数。 • url: 作成されたサポート バンドルの URL。 	JSONオブジェクト
duration	サポート バンドルの作成に使用された時間 (形式: HH:MM:SS.ssssss)。	string
結果	サポート バンドル操作の成功または失敗。	string

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "CreateSupportBundle",
  "params": {
    "extraArgs": "--compress gz"
  },
  "id": 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "details": {
      "bundleName": "supportbundle",
      "extraArgs": "--compress gz",
      "files": [
        "supportbundle.nodehostname.tar.gz"
      ],
      "output": "timeout -s KILL 1500s /sf/scripts/sfsupportbundle --quiet
--compress gz /tmp/solidfire-dtemp.1L6bdX/supportbundle<br><br>Moved
'/tmp/solidfire-dtemp.1L6bdX/supportbundle.nodehostname.tar.gz' to
/tmp/supportbundles",
      "timeoutSec": 1500,
      "url": [

        "https://nodeIP:442/config/supportbundles/supportbundle.nodehostname.tar.g
z"
      ]
    },
    "duration": "00:00:43.101627",
    "result": "Passed"
  }
}
```

バージョン以降の新機能

9.6

すべてのサポートバンドルを削除

使用することができます `DeleteAllSupportBundles` 生成されたすべてのサポートバンドルを削除する方法 `CreateSupportBundle` API メソッド。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドには戻り値はありません。

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "DeleteAllSupportBundles",
  "params": {}
},
"id": 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id" : 1,
  "result" : {}
}
```

バージョン以降の新機能

9.6

メンテナンスモードを無効にする

使用することができます `DisableMaintenanceMode` ストレージ ノードをメンテナンスモードから解除する方法。メンテナンスが完了し、ノードがオンラインになった後のみ、メンテナンス モードを無効にする必要があります。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
ノード	メンテナンス モードを解除するストレージ ノード ID のリスト。	整数配列	なし	はい

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
------	----	-----

非同期ハンドル	GetAsyncResult メソッドを使用してこの asyncHandle を取得し、メンテナンス モードの移行がいつ完了したかを判断できます。	integer
現在のモード	<p>ノードの現在のメンテナンス モードの状態。有効な値は次のとおりです。</p> <ul style="list-style-type: none"> • 無効: メンテナンスは要求されていません。 • FailedToRecover: ノードはメンテナンス モードからの回復に失敗しました。 • 予期しない: ノードはオフラインであることが判明しましたが、無効モードになっていました。 • RecoveringFromMaintenance: ノードはメンテナンス モードから回復中です。 • PreparingForMaintenance: ノードのメンテナンスを実行する準備をするためのアクションが実行されています。 • ReadyForMaintenance: ノードはメンテナンスを実行する準備ができています。 	メンテナンスモード (文字列)

要求モード	<p>要求されたノードのメンテナンスモードの状態。有効な値は次のとおりです。</p> <ul style="list-style-type: none"> • 無効: メンテナンスは要求されていません。 • FailedToRecover: ノードはメンテナンス モードからの回復に失敗しました。 • 予期しない: ノードはオフラインであることが判明しましたが、無効モードになっていました。 • RecoveringFromMaintenance: ノードはメンテナンス モードから回復中です。 • PreparingForMaintenance: ノードのメンテナンスを実行する準備をするためのアクションが実行されています。 • ReadyForMaintenance: ノードはメンテナンスを実行する準備ができています。 	メンテナンスモード (文字列)
-------	--	-----------------

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "DisableMaintenanceMode",
  "params": {
    "nodes": [6]
  },
  "id": 1
}
```

応答例

このメソッドは、次の例のような応答を返します。


```
{
  "id": 1,
  "result": {
    "requestedMode": "Disabled",
    "asyncHandle": 1,
    "currentMode": "Enabled"
  }
}
```

バージョン以降の新機能

12.2

詳細情報の参照

["NetApp HCIストレージメンテナンスモードの概念"](#)

SSHを無効にする

使用することができます `DisableSsh` 単一のストレージノードの SSH サービスを無効にする方法。この方法は、クラスター全体の SSH サービスのタイムアウト期間には影響しません。

パラメータ

このメソッドには入力パラメータがありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
有効	このノードの SSH サービスのステータス。	ブーリアン

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "DisableSsh",
  "params": {
    },
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id" : 1,
  "result" : {"enabled": false}
}
```

メンテナンスモードを有効にする

使用することができます `EnableMaintenanceMode` ストレージノードをメンテナンス用に準備する方法。メンテナンス シナリオには、ノードの電源をオフにしたり再起動したりする必要があるタスクが含まれます。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
未解決の障害を強制する	ブロッキング クラスタ障害が発生している場合でも、このノードに対して強制的にメンテナンスモードを有効にします。	ブーリアン	間違い	いいえ
ノード	メンテナンス モードにするノード ID のリスト。一度にサポートされるノードは 1 つだけです。	整数配列	なし	はい

Name	説明	タイプ	デフォルト値	必須
1分あたりのプライマリスワップ制限	1 分あたりにスワップするプライマリスライスの数。指定しない場合は、すべてのプライマリスライスが一度に交換されます。	integer	なし	いいえ
timeout	メンテナンス モードが自動的に無効になるまでの有効期間を指定します。時間文字列としてフォーマットされます (例: HH:mm:ss)。指定しない場合は、明示的に無効にするまでメンテナンス モードは有効のままになります。	string	なし	いいえ

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
非同期ハンドル	GetAsyncResult メソッドを使用してこの asyncHandle を取得し、メンテナンス モードの移行がいつ完了したかを判断できます。	integer

現在のモード	<p>ノードの現在のメンテナンス モードの状態。有効な値は次のとおりです。</p> <ul style="list-style-type: none"> • 無効: メンテナンスは要求されていません。 • FailedToRecover: ノードはメンテナンス モードからの回復に失敗しました。 • RecoveringFromMaintenance: ノードはメンテナンス モードから回復中です。 • PreparingForMaintenance: ノードのメンテナンスを実行する準備をするためのアクションが実行されています。 • ReadyForMaintenance: ノードはメンテナンスを実行する準備ができています。 	メンテナンスモード (文字列)
要求モード	<p>要求されたノードのメンテナンス モードの状態。有効な値は次のとおりです。</p> <ul style="list-style-type: none"> • 無効: メンテナンスは要求されていません。 • FailedToRecover: ノードはメンテナンス モードからの回復に失敗しました。 • RecoveringFromMaintenance: ノードはメンテナンス モードから回復中です。 • PreparingForMaintenance: ノードのメンテナンスを実行する準備をするためのアクションが実行されています。 • ReadyForMaintenance: ノードはメンテナンスを実行する準備ができています。 	メンテナンスモード (文字列)

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "EnableMaintenanceMode",
  "params": {
    "forceWithUnresolvedFaults": False,
    "nodes": [6],
    "perMinutePrimarySwapLimit" : 40,
    "timeout" : "01:00:05"
  },
  "id": 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result":
    {
      "requestedMode": "ReadyForMaintenance",
      "asyncHandle": 1,
      "currentMode": "Disabled"
    }
}
```

バージョン以降の新機能

12.2

詳細情報の参照

["NetApp HCIストレージメンテナンスモードの概念"](#)

SSHを有効にする

使用することができます `EnableSsh` 単一ノードに対してセキュア シェル (SSH) サービスを有効にする方法。この方法は、クラスター全体の SSH タイムアウト期間には影響せず、グローバル SSH タイムアウトによってノードの SSH が無効になることを免除しません。

パラメータ

このメソッドには入力パラメータがありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
有効	このノードの SSH サービスのステータス。	ブーリアン

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "EnableSsh",
  "params": {
  },
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id" : 1,
  "result" : {"enabled": true}
}
```

クラスタ構成の取得

使用することができます `GetClusterConfig` ノードがクラスターとの通信に使用するクラスター構成に関する情報を返す API メソッド。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
cluster	ノードがクラスターと通信するために使用するクラスター構成情報。	cluster

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "GetClusterConfig",
  "params": {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "cluster": {
      "cipi": "Bond10G",
      "cluster": "ClusterName",
      "ensemble": [
        "1:10.30.65.139",
        "2:10.30.65.140",
        "3:10.30.65.141"
      ],
      "fipsDriveConfiguration": true,
      "mipi": "Bond1G",
      "name": "xxx-en142",
      "nodeID": 4,
      "pendingNodeID": 0,
      "role": "Storage",
      "sipi": "Bond10G",
      "state": "Active",
      "version": "9.1.0"
    }
  }
}
```

クラスター状態を取得する

使用することができます `GetClusterState` ノードがクラスターの一部であるかどうかを示す API メソッド。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
cluster	クラスタの名前。	string
状態	<ul style="list-style-type: none">• 使用可能: ノードにクラスター名が設定されていません。• 保留中: ノードは特定の名前付きクラスターに対して保留中であり、追加できます。• アクティブ: ノードはクラスターのアクティブ メンバーであり、別のクラスターに追加することはできません。	string

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "GetClusterState",
  "params": {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。


```
{
  "id" : 1,
  "result" :
    "cluster" : "Cluster101"
    "state" : "Active"
}
```

バージョン以降の新機能

9.6

ゲットコンフィグ

使用することができます `GetConfig` ノードのすべての構成情報を取得するための API メソッド。この API メソッドには、`GetClusterConfig` そして `GetNetworkConfig` API メソッド。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
設定	<p>クラスターの構成の詳細。このオブジェクトには次のものが含まれます。</p> <ul style="list-style-type: none">• cluster: ストレージ ノードが関連付けられているストレージ クラスターと通信する方法を識別するクラスター情報。• ネットワーク (すべてのインターフェース): ノードの各ネットワーク インターフェイスのネットワーク接続タイプと現在の設定。	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "GetConfig",
  "params": {},
  "id" : 1
}
```

応答例

この応答例は長いため、補足トピックに記載されています。

バージョン以降の新機能

9.6

詳細情報の参照

- [クラスタ構成の取得](#)
- [ネットワーク構成の取得](#)
- [ゲットコンフィグ](#)

ドライブ構成を取得

使用することができます `GetDriveConfig` 予想されるスライスおよびブロック ドライブ 数と、現在ノードに接続されているスライスおよびブロック ドライブの数に関するドライブ情報を取得するメソッド。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
ドライブ構成	ノードに接続されているドライブに関する情報。	ドライブ

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{  
  "method": "GetDriveConfig",  
  "params": {},  
  "id" : 1  
}
```

応答例

このメソッドの応答は次の例のようになります。長さのため、応答には1つのストレージノードの1つのドライブの情報のみが含まれます。

```

{
  "id": 1,
  "result": {
    "driveConfig": {
      "drives": [
        {
          "canonicalName": "sda",
          "connected": true,
          "dev": 2052,
          "devPath": "/dev/sdimm0p4",
          "driveType": "Slice",
          "name": "scsi-SATA_VRFSD3400GNCVMT205581853-
part4",
          "path": "/dev/sda4",
          "pathLink": "/dev/sdimm0p4",
          "product": "VRFSD3400GNCVMTKS1",
          "scsiCompatId": "scsi-
SATA_VRFSD3400GNCVMT205581853-part4",
          "scsiState": "Running",
          "securityAtMaximum": false,
          "securityEnabled": false,
          "securityFrozen": true,
          "securityLocked": false,
          "securitySupported": true,
          "serial": "205581853",
          "size": 299988156416,
          "slot": -1,
          "uuid": "9d4b198b-5ff9-4f7c-04fc-
3bc4e2f38974",
          "vendor": "Viking",
          "version": "612ABBF0"
        }
      ],
      "numBlockActual": 10,
      "numBlockExpected": 10,
      "numSliceActual": 1,
      "numSliceExpected": 1,
      "numTotalActual": 11,
      "numTotalExpected": 11
    }
  }
}

```

ハードウェア構成の取得

使用することができます `GetHardwareConfig` ノードのハードウェア構成情報を取得するメソッド。この構成データは内部使用を目的としています。より便利なライブシステムハードウェアコンポーネントインベントリを取得するには、`GetHardwareInfo` 代わりにメソッドを使用します。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
hardwareConfig	ハードウェア情報と現在の設定のリスト。	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "GetHardwareConfig",
  "params": {},
  "id" : 1
}
```

応答例

このメソッドの応答は次の例のようになります。

```
{
  "id": 1,
  "result": {
    "hardwareConfig": {
      "biosRevision": "1.0",
      "biosVendor": [
        "NetApp",
        "SolidFire"
      ],
      "biosVersion": "1.1.2",
      "blockDriveSizeBytes": 300069052416,

```

```

"blockDrives": [
    "/dev/slot0",
    "/dev/slot1",
    "/dev/slot2",
    "/dev/slot3",
    "/dev/slot4",
    "/dev/slot5",
    "/dev/slot6",
    "/dev/slot7",
    "/dev/slot8",
    "/dev/slot9"
],
"blockServiceFormat": "Standard",
"bmcFirmwareRevision": "1.6",
"bmcIpmiVersion": "2.0",
"chassisType": "R620",
"cpuCores": 6,
"cpuCoresEnabled": 6,
"cpuModel": "Intel(R) Xeon(R) CPU E5-2640 0 @ 2.50GHz",
"cpuThreads": 12,
"driveSizeBytesInternal": 400088457216,
"fibreChannelFirmwareRevision": "",
"fibreChannelModel": "",
"fibreChannelPorts": {},
"idracVersion": "1.06.06",
"ignoreFirmware": [],
"memoryGB": 72,
"memoryMhz": 1333,
"networkDriver": [
    "bnx2x"
],
"nicPortMap": {
    "PortA": "eth2",
    "PortB": "eth3",
    "PortC": "eth0",
    "PortD": "eth1"
},
"nodeType": "SF3010",
"numCpu": 2,
"numDrives": 10,
"numDrivesInternal": 1,
"nvramTempMonitorEnable": false,
"rootDrive": "/dev/sdimm0",
"scsiBusExternalDriver": "mpt3sas",
"scsiBusInternalDriver": "ahci",
"sliceDriveSizeBytes": 299988156416,

```

```

        "sliceDrives": [
            "/dev/sdimm0p4"
        ],
        "slotOffset": 0,
        "solidfireDefaults": {
            "bufferCacheGB": 12,
            "configuredIops": 50000,
            "cpuDmaLatency": -1,
            "driveWriteThroughputMBPerSleep": 10,
            "maxDriveWriteThroughputMBPerSec": 175,
            "maxIncomingSliceSyncs": 10,
            "postCallbackThreadCount": 8,
            "sCacheFileCapacity": 100000000,
            "sliceFileLogFileCapacity": 5000000000
        }
    }
}

```

バージョン以降の新機能

9.6

ハードウェア情報を取得する

使用することができます `GetHardwareInfo` 単一ノードのライブハードウェア情報とステータスを取得する方法。ハードウェア情報には通常、製造元、ベンダー、バージョン、ドライブ、その他の関連する識別情報が含まれます。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
カ	クラスター内のすべてのノードで実行するには、この「force」パラメータをtrue に設定します。	ブーリアン	false	いいえ

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
ハードウェア情報	ノードのハードウェア情報。	ハードウェア情報

リクエスト例

このメソッドのリクエストは次の例ようになります。

```
{
  "method": "GetHardwareInfo",
  "params": {
  },
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "hardwareInfo": {
      "bus": {
        "core_DMI:0200": {
          "description": "Motherboard",
          "physid": "0",
          "product": "0A47AA",
          "serial": "..AB123456C12354.",
          "version": "C07"
        }
      },
      "driveHardware": [
        {
          "canonicalName": "sdh",
          "connected": true,
          "dev": 2160,
          "devPath": "/dev/disk/by-path/pci-0000:41:00.0-sas-0x500056b37789abf0-lun-0",
          "driveEncryptionCapability": "fips",
          "driveType": "Block",
          "lifeRemainingPercent": 92,
          "lifetimeReadBytes": 175436696911872,
          "lifetimeWriteBytes": 81941097349120,

```



```

    "name": "scsi-SATA_INTEL_SSDSC2BB3BTWL12345686300AAA",
    "path": "/dev/sdh",
    "pathLink": "/dev/disk/by-path/pci-0000:41:00.0-sas-
0x500056b37789abf0-lun-0",
    "powerOnHours": 17246,
    "product": "INTEL SSDAA2AA300A4",
    "reallocatedSectors": 0,
    "reserveCapacityPercent": 100,
    "scsiCompatId": "scsi-SATA_INTEL_SSDSC2BB3BTWL12345686300AAA",
    "scsiState": "Running",
    "securityAtMaximum": false,
    "securityEnabled": false,
    "securityFrozen": false,
    "securityLocked": false,
    "securitySupported": true,
    "serial": "AAAA33710886300AAA",
    "size": 300069052416,
    "slot": 1,
    "smartSsdWriteCapable": false,
    "uuid": "aea178b9-c336-6bab-a61d-87b615e8120c",
    "vendor": "Intel",
    "version": "D2010370"
  },
  ...
]
}
}
}

```

バージョン以降の新機能

9.6

GetIpmiConfig

使用することができます `GetIpmiConfig` ノード内のセンサーからハードウェア センサー情報を取得するメソッド。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ
シャーシタイプ	<p>各ノード シャーシ タイプの情報を表示するために使用されます。有効な値は次のとおりです。</p> <ul style="list-style-type: none"> • all: 各シャーシ タイプのセンサー情報を返します。 • {シャーシ タイプ}: 指定されたシャーシ タイプのセンサー情報を返します。 	string

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
センサー名	見つかったセンサーの名前。	string
ユニークセンサーID	センサーの一意的識別子。	string

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "GetIpmiConfig",
  "params": {
    "chassisType": "all"
  },
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "nodes": [
      {
        "nodeID": 1,
        "result": {
```

```

"ipmiConfig": {
  "C220M4": [
    {
      "sensorName": "Fan1A RPM",
      "uniqueSensorID": "29.1:0xf"
    },
    {
      "sensorName": "Fan1B RPM",
      "uniqueSensorID": "29.1:0x10"
    },
    {
      "sensorName": "Fan2A RPM",
      "uniqueSensorID": "29.2:0x11"
    },
    {
      "sensorName": "Fan2B RPM",
      "uniqueSensorID": "29.2:0x12"
    },
    {
      "sensorName": "Fan3A RPM",
      "uniqueSensorID": "29.3:0x13"
    },
    {
      "sensorName": "Fan3B RPM",
      "uniqueSensorID": "29.3:0x14"
    },
    {
      "sensorName": "Fan4A RPM",
      "uniqueSensorID": "29.4:0x15"
    },
    {
      "sensorName": "Fan4B RPM",
      "uniqueSensorID": "29.4:0x16"
    },
    {
      "sensorName": "Fan5A RPM",
      "uniqueSensorID": "29.5:0x17"
    },
    {
      "sensorName": "Fan5B RPM",
      "uniqueSensorID": "29.5:0x18"
    },
    {
      "sensorName": "Fan6A RPM",
      "uniqueSensorID": "29.6:0x19"
    },
  ],

```

```

    {
      "sensorName": "Fan6B RPM",
      "uniqueSensorID": "29.6:0x1a"
    },
    {
      "sensorName": "Exhaust Temp",
      "uniqueSensorID": "7.1:0x1"
    },
    {
      "sensorName": "Inlet Temp",
      "uniqueSensorID": "7.1:0x4"
    },
    {
      "sensorName": "PS1",
      "uniqueSensorID": "10.1:0x26"
    },
    {
      "sensorName": "PS2",
      "uniqueSensorID": "10.2:0x2c"
    }
  ],
  "R620": [
    {
      "sensorName": "Fan1A RPM",
      "uniqueSensorID": "7.1:0x30"
    },
    {
      "sensorName": "Fan1B RPM",
      "uniqueSensorID": "7.1:0x31"
    },
    {
      "sensorName": "Fan2A RPM",
      "uniqueSensorID": "7.1:0x32"
    },
    {
      "sensorName": "Fan2B RPM",
      "uniqueSensorID": "7.1:0x33"
    },
    {
      "sensorName": "Fan3A RPM",
      "uniqueSensorID": "7.1:0x34"
    },
    {
      "sensorName": "Fan3B RPM",
      "uniqueSensorID": "7.1:0x35"
    },
  ],

```

```

{
  "sensorName": "Fan4A RPM",
  "uniqueSensorID": "7.1:0x36"
},
{
  "sensorName": "Fan4B RPM",
  "uniqueSensorID": "7.1:0x37"
},
{
  "sensorName": "Fan5A RPM",
  "uniqueSensorID": "7.1:0x38"
},
{
  "sensorName": "Fan5B RPM",
  "uniqueSensorID": "7.1:0x39"
},
{
  "sensorName": "Fan6A RPM",
  "uniqueSensorID": "7.1:0x3a"
},
{
  "sensorName": "Fan6B RPM",
  "uniqueSensorID": "7.1:0x3b"
},
{
  "sensorName": "Fan7A RPM",
  "uniqueSensorID": "7.1:0x3c"
},
{
  "sensorName": "Fan7B RPM",
  "uniqueSensorID": "7.1:0x3d"
},
{
  "sensorName": "Exhaust Temp",
  "uniqueSensorID": "7.1:0x1"
},
{
  "sensorName": "Inlet Temp",
  "uniqueSensorID": "7.1:0x4"
},
{
  "sensorName": "PS1",
  "uniqueSensorID": "10.1:0x62"
},
{
  "sensorName": "PS2",

```

```
        "uniqueSensorID": "10.2:0x63"
      }
    ],
  }
}
```

バージョン以降の新機能

9.6

GetIpmiInfo

使用することができます `GetIpmiInfo` システムによって監視されるノード ファン、吸気および排気温度、および電源のセンサー (オブジェクト) の詳細なレポートを表示する方法。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
センサー	ノード内の各センサーからの詳細情報。	JSONオブジェクト配列

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "GetIpmiInfo",
  "params": {},
  "id" : 1
}
```

応答例

この API メソッドで返される応答は長いため、このドキュメントでは応答の一部が意図的に削除されています。含まれるのは、ノードが最適なパフォーマンスで実行されていることを確認するためにシステムが監視するハードウェア情報の一部です。

```
{
```

```

"id": 1,
  "result": {
    "ipmiInfo": {
      "sensors": [
        {
          "entityID": "7.1 (System Board)",
          "sensorID": "0x72",
          "sensorName": "SEL",
          "sensorType": "Event Logging Disabled",
          "uniqueSensorID": "7.1:0x72"
        },
        {
          "assertionsEnabled": [ "General Chassis intrusion" ],
          "deassertionsEnabled": [ "General Chassis intrusion" ],
          "entityID": "7.1 (System Board)", "sensorID": "0x73",
          "sensorName": "Intrusion",
          "sensorType": "Physical Security",
          "uniqueSensorID": "7.1:0x73"
        },
        {THIS ENTIRE SECTION IS REPEATED FOR EACH FAN IN THE SYSTEM
          "assertionEvents": [],
          "assertionsEnabled": [],
          "deassertionsEnabled": [],
          "entityID": "7.1 (System Board)",
          "eventMessageControl": "Per-threshold",
          "lowerCritical": "720.000",
          "lowerNonCritical": "840.000",
          "maximumSensorRange": "Unspecified",
          "minimumSensorRange": "Unspecified",
          "negativeHysteresis": "600.000",
          "nominalReading": "10080.000",
          "normalMaximum": "23640.000",
          "normalMinimum": "16680.000",
          "positiveHysteresis": "600.000",
          "readableThresholds": "lcr lnc",
          "sensorID": "0x30",
          "sensorName": "Fan1A RPM",
          "sensorReading": "4440 (+/- 120) RPM",
          "sensorType": "Fan",
          "settableThresholds": "",
          "status": "ok",
          "thresholdReadMask": "lcr lnc",
          "uniqueSensorID": "7.1:0x30"
        },
        .
        .

```

.
{THIS ENTIRE SECTION IS REPEATED FOR THE EXHAUST TEMPERATURE
OF EACH NODE

```
"assertionEvents": [],  
"assertionsEnabled": [],  
"entityID": "7.1 (System Board)",  
"eventMessageControl": "Per-threshold",  
"lowerCritical": "3.000",  
"lowerNonCritical": "8.000",  
"maximumSensorRange": "Unspecified",  
"minimumSensorRange": "Unspecified",  
"negativeHysteresis": "1.000",  
"nominalReading": "23.000",  
"normalMaximum": "69.000",  
"normalMinimum": "11.000",  
"positiveHysteresis": "1.000",  
"readableThresholds": "lcr lnc unc ucr",  
"sensorID": "0x1",  
"sensorName": "Exhaust Temp",  
"sensorReading": "44 (+/- 1) degrees C",  
"sensorType": "Temperature",  
"settableThresholds": "",  
"status": "ok",  
"uniqueSensorID": "7.1:0x1",  
"upperCritical": "75.000",  
"upperNonCritical": "70.000"  
},
```

{THIS ENTIRE SECTION IS REPEATED FOR THE INLET TEMPERATURE OF
EACH NODE

```
"assertionEvents": [],  
"assertionsEnabled": [],  
"deassertionsEnabled": [],  
"entityID": "7.1 (System Board)",  
"eventMessageControl": "Per-threshold",  
"lowerCritical": "-7.000",  
"lowerNonCritical": "3.000",  
"maximumSensorRange": "Unspecified",  
"minimumSensorRange": "Unspecified",  
"negativeHysteresis": "1.000",  
"nominalReading": "23.000",  
"normalMaximum": "69.000",  
"normalMinimum": "11.000",  
"positiveHysteresis": "1.000",  
"readableThresholds": "lcr lnc unc ucr",  
"sensorID": "0x4",  
"sensorName": "Inlet Temp",
```



```

        "sensorReading": "20 (+/- 1) degrees C",
        "sensorType": "Temperature",
        "settableThresholds": "lcr lnc unc ucr",
        "status": "ok",
        "thresholdReadMask": "lcr lnc unc ucr",
        "uniqueSensorID": "7.1:0x4",
        "upperCritical": "47.000",
        "upperNonCritical": "42.000"
    },
    {THIS ENTIRE SECTION IS REPEATED FOR EACH POWER SUPPLY ON EACH
NODE
        "assertionEvents": [],
        "assertionsEnabled": [],
        "entityID": "10.2 (Power Supply)",
        "eventMessageControl": "Per-threshold",
"maximumSensorRange": "Unspecified",
        "minimumSensorRange": "Unspecified",
        "negativeHysteresis": "Unspecified",
        "nominalReading": "0.000",
        "normalMaximum": "0.000",
        "positiveHysteresis": "Unspecified",
        "readableThresholds": "No Thresholds",
        "sensorID": "0x6d",
        "sensorName": "Voltage 2",
        "sensorReading": "118 (+/- 0) Volts",
        "sensorType": "Voltage",
        "settableThresholds": "No Thresholds", "status": "ok",
"uniqueSensorID": "10.2:0x6d"
    },
    .
    .
    .
    }
    ]
    }
    }
    }
    }

```

バージョン以降の新機能

9.6

ネットワーク構成の取得

使用することができます `GetNetworkConfig` ノードのネットワーク構成情報を表示する

メソッド。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
network	ノードの各ネットワーク インターフェイスのネットワーク接続タイプと現在の設定。	ネットワーク (すべてのインターフェイス)

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "GetNetworkConfig",
  "params": {},
  "id" : 1
}
```

応答例

この応答例は長いため、補足トピックに記載されています。

バージョン以降の新機能

9.6

詳細情報の参照

[ネットワーク構成の取得](#)

ネットワークインターフェースの取得

使用することができます `GetNetworkInterface` ノード上のネットワーク インターフェイスに関する情報を取得するメソッド。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
インタフェース	各ノードの情報を取得するインターフェースの名前。有効な値は次のとおりです。 <ul style="list-style-type: none"> • ボンド1G • ボンド10G 	string	なし	いいえ
力	クラスター内のすべてのノードで実行するには、このパラメータを true に設定します。	ブーリアン	false	いいえ

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
ノード	ストレージ クラスター内の各ストレージ ノードのインターフェイスを記述するオブジェクトの配列。配列内の各オブジェクトには、次の項目が含まれます。 <ul style="list-style-type: none"> • nodeID: (整数) インターフェイス情報が適用されるストレージ クラスター内のストレージ ノードの ID。 • 結果: (ネットワークインターフェース) このストレージ ノードのインターフェイス構成情報。 	JSONオブジェクト配列

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "GetNetworkInterface",
  "params": {
    "interface": "Bond1G",
    "force": true
  },
  "id": 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "nodes": [
      {
        "nodeID": 1,
        "result": {
          "interface": {
            "address": "10.117.64.32",
            "addressV6": ":::",
            "broadcast": "10.117.79.255",
            "macAddress": "90:b1:1c:42:e0:1e",
            "mtu": 1500,
            "name": "Bond1G",
            "namespace": false,
            "netmask": "255.255.240.0",
            "status": "UpAndRunning",
            "type": "BondMaster",
            "virtualNetworkTag": 0
          }
        }
      },
      {
        "nodeID": 2,
        "result": {
          "interface": {
            "address": "10.117.64.35",
            "addressV6": ":::",
            "broadcast": "10.117.79.255",
            "macAddress": "d4:ae:52:7a:ae:23",
            "mtu": 1500,
```

```

        "name": "Bond1G",
        "namespace": false,
        "netmask": "255.255.240.0",
        "status": "UpAndRunning",
        "type": "BondMaster",
        "virtualNetworkTag": 0
    }
}
},
{
    "nodeID": 3,
    "result": {
        "interface": {
            "address": "10.117.64.39",
            "addressV6": "::",
            "broadcast": "10.117.79.255",
            "macAddress": "c8:1f:66:f0:9d:17",
            "mtu": 1500,
            "name": "Bond1G",
            "namespace": false,
            "netmask": "255.255.240.0",
            "status": "UpAndRunning",
            "type": "BondMaster",
            "virtualNetworkTag": 0
        }
    }
},
{
    "nodeID": 4,
    "result": {
        "interface": {
            "address": "10.117.64.107",
            "addressV6": "::",
            "broadcast": "10.117.79.255",
            "macAddress": "b8:ca:3a:f5:24:f8",
            "mtu": 1500,
            "name": "Bond1G",
            "namespace": false,
            "netmask": "255.255.240.0",
            "status": "UpAndRunning",
            "type": "BondMaster",
            "virtualNetworkTag": 0
        }
    }
}
}
]

```

```
}  
}
```

バージョン以降の新機能

9.6

GetNodeActiveTlsCiphers

使用することができます `GetNodeActiveTlsCiphers` 単一ノードでメソッドを実行して、このノードで現在受け入れられている TLS 暗号のリストを取得します。この方法は、管理ノードとストレージノードで使用できます。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
必須暗号	ノードの必須 TLS 暗号スイートのリスト。これらはノード上で常にアクティブな暗号です。	string
補足暗号	ノードの補足 TLS 暗号スイートのリスト。	string

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{  
  "method": "GetNodeActiveTlsCiphers",  
  "params": {},  
  "id" : 1  
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id" : 1,
  "result" : {
    "mandatoryCiphers": [
      "DHE-RSA-AES256-SHA256",
      "DHE-RSA-AES256-GCM-SHA384",
      "ECDHE-RSA-AES256-SHA384",
      "ECDHE-RSA-AES256-GCM-SHA384"
    ],
    "supplementalCiphers": [
      "DHE-RSA-AES128-SHA256",
      "DHE-RSA-AES128-GCM-SHA256",
      "ECDHE-RSA-AES128-SHA256",
      "ECDHE-RSA-AES128-GCM-SHA256"
    ]
  }
}
```

GetNodeFipsDrivesReport

使用することができます `GetNodeFipsDrivesReport` ストレージ クラスター内の単一ノードの FIPS 140-2 ドライブ暗号化機能のステータスを確認する方法。このメソッドは、個々のストレージ ノードに対して実行する必要があります。

パラメータ

このメソッドには入力パラメータがありません。

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
fipsドライブ	<p>このノードの FIPS 140-2 機能のサポートのステータスを含む JSON オブジェクト。有効な値は次のとおりです。</p> <ul style="list-style-type: none"> なし: ノードは FIPS に対応していません。 部分的: ノードは FIPS 対応ですが、ノード内のすべてのドライブが FIPS ドライブではありません。 準備完了: ノードは FIPS 対応であり、ノード内のすべてのドライブは FIPS ドライブです (またはドライブが存在しません)。 	string

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "GetNodeFipsDrivesReport",
  "params": {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "fipsDrives": "None"
  }
}
```

バージョン以降の新機能

11.5

ノードSSL証明書の取得

使用することができます `GetNodeSSLCertificate` 管理ノードで現在アクティブな SSL 証明書を取得する方法。

パラメータ



このメソッドは管理ノードに対して呼び出す必要があります。例えば：

```
https://<management node IP>:442/json-rpc/10.0
```

このメソッドには入力パラメータはありません。

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
証明書	証明書の完全な PEM エンコードされたテキスト。	string
詳細	証明書のデコードされた情報。	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例ようになります。

```
{
  "method" : "GetNodeSSLCertificate",
  "params" : {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "certificate": "-----BEGIN CERTIFICATE-----
\nMIIEdzCCA1+gAwIBAgIJAMwbIhWY43/zMA0GCSqGSIb3DQEBBQUAMIGDMQswCQYD\nVQZGGEw
```


GetNodeSupportedTlsCiphers

使用することができます `GetNodeSupportedTlsCiphers` 単一ノードでメソッドを実行して、このノードで現在サポートされている TLS 暗号のリストを取得します。この方法は、管理ノードとストレージノードで使用できます。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
必須暗号	ノードの必須 TLS 暗号スイートのリスト。これらはノード上で常にアクティブな暗号です。	string
デフォルトの補足暗号	ノードのデフォルトの補足 TLS 暗号スイートのリスト。 ResetNodeSupplementalTlsCiphers API メソッドを実行すると、補足暗号がこのリストに復元されます。	string
サポートされている補足暗号	SetNodeSupplementalTlsCiphers API メソッドで設定できる利用可能な補足 TLS 暗号スイートのリスト。	string

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "GetNodeSupportedTlsCiphers",
  "params": {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```

{
  "id" : 1,
  "result" : {
    "defaultSupplementalCiphers": [
      "DHE-RSA-AES128-SHA256",
      "DHE-RSA-AES128-GCM-SHA256",
      "ECDHE-RSA-AES128-SHA256",
      "ECDHE-RSA-AES128-GCM-SHA256"
    ],
    "mandatoryCiphers": [
      "DHE-RSA-AES256-SHA256",
      "DHE-RSA-AES256-GCM-SHA384",
      "ECDHE-RSA-AES256-SHA384",
      "ECDHE-RSA-AES256-GCM-SHA384"
    ],
    "supportedSupplementalCiphers": [
      "DHE-RSA-AES128-SHA256",
      "DHE-RSA-AES128-GCM-SHA256",
      "ECDHE-RSA-AES128-SHA256",
      "ECDHE-RSA-AES128-GCM-SHA256",
      "DHE-RSA-AES256-SHA",
      "ECDHE-RSA-AES256-SHA",
      "DHE-RSA-CAMELLIA256-SHA",
      "DHE-RSA-AES128-SHA",
      "ECDHE-RSA-AES128-SHA",
      "DHE-RSA-CAMELLIA128-SHA"
    ]
  }
}

```

パッチ情報を取得

使用することができます `GetPatchInfo` ストレージ ノードにインストールされている Element ソフトウェア パッチに関する情報を取得する方法。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
force	<p>ストレージ クラスター内のすべてのノードでメソッドを強制的に実行します。これは、単一のノードではなくクラスター IP アドレスに API を発行する場合にのみ必要です。有効な値は次のとおりです。</p> <ul style="list-style-type: none"> • true • false 	ブーリアン	false	いいえ

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
パッチ	このノードにインストールされているパッチに関する情報を含むオブジェクト。	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "GetPatchInfo",
  "params": {
    "force": false,
  },
  "id": 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "patches": {
      "SUST936": {
        "date": "Wed 09 Dec 2020 10:41:59 PM UTC",
        "description": "BMC fixes",
        "newFiles": [
          "None"
        ],
        "patchedFiles": [
          "Patched_file_1.bin",
          "Patched_file_2.dat",
          "Patched_file_3.tgz"
        ]
      }
    }
  }
}
```

バージョン以降の新機能

12.3

保留中の操作を取得する

使用することができます `GetPendingOperation` 現在進行中のノード上の操作を検出する方法。このメソッドは、操作が完了したときに報告するためにも使用できます。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
pending	<p>有効な値は次のとおりです。</p> <ul style="list-style-type: none"> • true: 操作はまだ進行中です。 • 間違い： <p>操作は進行中ではありません。</p>	ブーリアン
operation	進行中または完了した操作の名前。	string

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "GetPendingOperation",
  "params": {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id" : 1,
  "result" : {
    "pendingOperation" : {
      "pending" : "true",
      "operation" : "TestDrivesInternal",
    }
  }
}
```

バージョン以降の新機能

9.6

GetSshInfo

使用することができます `GetSshInfo` 単一ノード上の SSH サービスのステータスを照会

する方法。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
結果	このノードの SSH サービスのステータス。	ブーリアン

リクエスト例

このメソッドのリクエストは次の例ようになります。

```
{
  "method" : "GetSshInfo",
  "params" : {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "enabled": false
  }
}
```

ドライブハードウェアのリスト

使用することができます `ListDriveHardware` ノードに接続されているすべてのドライブを一覧表示するメソッド。個々のノードで使用すると、このメソッドはドライブのハードウェア情報を返します。クラスター マスター ノード MVIP で使用すると、このメソッドはすべてのノード上のすべてのドライブの情報を返します。

パラメータ



メソッド応答の「securitySupported」: true 行は、ドライブが暗号化可能であることを意味するのではなく、セキュリティ ステータスを照会できることだけを意味します。モデル番号が「-NE」で終わるノード タイプがある場合、これらのドライブのセキュリティ機能を有効にするコマンドは失敗します。

このメソッドには次のパラメータがあります。

Name	説明	タイプ	デフォルト値	必須
力	このメソッドをすべてのノードで実行するには、true に設定します。	ブーリアン	なし	いいえ

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
ドライブハードウェア	ノードのドライブ ハードウェア情報が返されました。	JSONオブジェクト配列

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "ListDriveHardware",
  "params": {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```

{
  "id": 1,
  "result": {
    "driveHardware": [
      {
        "canonicalName": "sda",
        "connected": true,
        "dev": 2048,
        "devPath": "/dev/slot0",
        "driveEncryptionCapability": "fips",
        "driveType": "Slice",
        "lifeRemainingPercent": 98,
        "lifetimeReadBytes": 0,
        "lifetimeWriteBytes": 14012129542144,
        "name": "scsi-SATA_SAMSUNG_MZ7GE24S1M9NWAG501251",
        "path": "/dev/sda",
        "pathLink": "/dev/slot0",
        "powerOnHours": 15489,
        "product": "SAMSUNG MZ7GE240HMGR-00003",
        "reallocatedSectors": 0,
        "reserveCapacityPercent": 100,
        "scsiCompatId": "scsi-SATA_SAMSUNG_MZ7GE24S1M9NWAG501251",
        "scsiState": "Running",
        "securityAtMaximum": false,
        "securityEnabled": true,
        "securityFrozen": false,
        "securityLocked": false,
        "securitySupported": true,
        "serial": "S1M9NWAG501251",
        "size": 240057409536,
        "slot": 0,
        "uncorrectableErrors": 0,
        "uuid": "789aa05d-e49b-ff4f-f821-f60eed8e43bd",
        "vendor": "Samsung",
        "version": "EXT1303Q"
      }
    ]
  }
}

```

バージョン以降の新機能

9.6

ネットワークインターフェースの一覧

使用することができます `ListNetworkInterfaces` ノード上の各ネットワーク インターフェイスに関する情報を一覧表示するメソッド。この API メソッドは個々のノードで使用することを目的としています。個々のノードにアクセスするには、ユーザー ID とパスワードの認証が必要です。ただし、メソッド呼び出しでパラメーター `force` に `true` の値が指定されている場合は、クラスターでこのメソッドを使用できます。パラメータをクラスターで使用すると、すべてのインターフェースがリストされます。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
力	有効な値は次のとおりです。 <ul style="list-style-type: none">• <code>true</code>: クラスター内のすべてのネットワーク インターフェイスに関する情報が返されます。• <code>false</code>: 情報は返されません。	ブーリアン	なし	いいえ

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
インターフェース	ストレージノード（または、 <code>force = true</code> の場合はストレージクラスタ全体）の各ネットワークインターフェースの構成情報のリスト。	ネットワークインターフェース配列

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "ListNetworkInterfaces",
  "params": {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "nodes": [
      {
        "nodeID": 1,
        "result": {
          "interfaces": [
            {
              "address": "10.117.80.32",
              "addressV6": "::",
              "broadcast": "10.117.95.255",
              "macAddress": "90:b1:1c:42:e0:1a",
              "mtu": 9000,
              "name": "Bond10G",
              "namespace": false,
              "netmask": "255.255.240.0",
              "status": "UpAndRunning",
              "type": "BondMaster",
              "virtualNetworkTag": 0
            },
            {
              "address": "10.117.64.32",
              "addressV6": "::",
              "broadcast": "10.117.79.255",
              "macAddress": "90:b1:1c:42:e0:1e",
              "mtu": 1500,
              "name": "Bond1G",
              "namespace": false,
              "netmask": "255.255.240.0",
              "status": "UpAndRunning",
              "type": "BondMaster",
              "virtualNetworkTag": 0
            }
          ]
        }
      }
    ]
  }
}
```

```

{
  "address": "0.0.0.0",
  "addressV6": "::",
  "broadcast": "0.0.0.0",
  "macAddress": "90:b1:1c:42:e0:1a",
  "mtu": 9000,
  "name": "eth0",
  "namespace": false,
  "netmask": "0.0.0.0",
  "status": "UpAndRunning",
  "type": "BondSlave",
  "virtualNetworkTag": 0
},
{
  "address": "127.0.0.1",
  "addressV6": "::",
  "broadcast": "0.0.0.0",
  "macAddress": "00:00:00:00:00:00",
  "mtu": 0,
  "name": "lo",
  "namespace": false,
  "netmask": "0.0.0.0",
  "status": "UpAndRunning",
  "type": "Loopback",
  "virtualNetworkTag": 0
}
]
}
}
]
}
}
}

```

バージョン以降の新機能

9.6

ネットワークインターフェース統計の一覧

使用することができます。`ListNetworkInterfaceStats` ノード上の各ネットワーク インターフェイスのドロップされたパケットの数やさまざまな種類のエラーなどの統計情報を一覧表示するメソッド。この API メソッドは個々のノードで使用することを目的としています。個々のノードにアクセスするには、ユーザー ID とパスワードの認証が必要です。ただし、メソッド呼び出しでパラメーター `force` に `true` の値が指定されている場合は、クラスターでこのメソッドを使用できます。パラメータをクラスターで使用すると、す

すべてのインターフェースのネットワーク統計が一覧表示されます。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
ネットワークインターフェース統計	ストレージ ノードの各ネットワーク インターフェイスの、ドロップされたパケットの数やさまざまな種類のネットワーク エラーなどのネットワーク統計情報のリスト。	ネットワークインターフェース統計配列

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "ListNetworkInterfaceStats",
  "params": {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "networkInterfaceStats": [
    {
      "rxErrors": 1,
      "rxPackets": 1,
      "txErrors": 1,
      "rxDropped": 1,
      "txCarrierErrors": 1,
      "rxOverErrors": 1,
      "rxMissedErrors": 1,
      "txPackets": 1,
      "name": "if_name",
      "rxLengthErrors": 1,
      "collisions": 1,
      "rxFifoErrors": 1,
      "txBytes": 1,
      "rxBytes": 1,
      "rxFrameErrors": 1,
      "rxCrcErrors": 1,
      "txFifoErrors": 1
    }
  ]
}
```

バージョン以降の新機能

12.3

リストテスト

使用することができます `ListTests` ノード上で実行できるテストを一覧表示するメソッド。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
テスト	ノードで実行できるテストのリスト。	文字列配列

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "ListTests",
  "params": {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "tests": [
      "TestConnectEnsemble",
      "TestConnectMvip",
      "TestConnectSvip",
      "TestDrives",
      "TestHardwareConfig",
      "TestLocateCluster",
      "TestPing",
      "TestLocalConnectivity",
      "TestRemoteConnectivity",
      "TestNetworkConfig"
    ]
  }
}
```

バージョン以降の新機能

9.6

リステューティリティ

使用することができます `ListUtilities` ノード上で実行できる操作を一覧表示するメソッド。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
ユーティリティ	現在ノード上で実行できるユーティリティのリスト。	文字列配列

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "ListUtilities",
  "params": {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "utilities": [
      "ResetDrives",
      "ResetNode",
      "RestartNetworking",
      "RestartServices",
      "CreateSupportBundle",
      "DeleteAllSupportBundles",
      "CreateClusterSupportBundle"
    ]
  }
}
```

バージョン以降の新機能

9.6

ノードSSL証明書の削除

使用することができます `RemoveNodeSSLCertificate` 管理ノードのユーザー SSL 証明書と秘密キーを削除する方法。証明書と秘密キーが削除されると、管理ノードはデフォルトの証明書と秘密キーを使用するように構成されます。

パラメータ



このメソッドは管理ノードに対して呼び出す必要があります。例えば：

```
https://<management node IP>:442/json-rpc/10.0
```

このメソッドには入力パラメータはありません。

戻り値

このメソッドには戻り値はありません。

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method" : "RemoveNodeSSLCertificate",
  "params" : {},
  "id" : 3
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id" : 3,
  "result" : {}
}
```

リセットドライブ

使用することができます `ResetDrives` ドライブを事前に初期化し、現在ドライブ上にあるすべてのデータを削除する方法。ドライブは既存のノードで再利用したり、アップグレードされたノードで使用したりできます。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
ドライブ	リセットするデバイス名 (ドライブ ID ではない) のリスト。	string	なし	はい
力	ドライブをリセットするには true に設定します。	ブーリアン	なし	はい

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
詳細	リセットされるドライブの詳細。	JSONオブジェクト配列

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "ResetDrives",
  "params": {
    "drives" : "slot3",
    "force" : true
  },
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "details": {
      "drives": [
        {
          "drive": "slot3",
          "returnCode": 0,
          "stderr": " * Unlocking /dev/slot9 .[ ok ]\ * Setting master
password /dev/slot9 .[ ok ]\ * Secure erasing /dev/slot9 (hdparm)
[tries=0/1] .....[ ok ]",
          "stdout": ""
        }
      ]
    },
    "duration": "00:00:28.501269",
    "result": "Passed"
  }
}
```

バージョン以降の新機能

9.6

リセットノード

使用することができます `ResetNode` ノードを工場出荷時の設定にリセットするメソッド。このメソッドを呼び出すと、すべてのデータ、パッケージ（ソフトウェア アップグレードなど）、設定、およびログ ファイルがノードから削除されます。ただし、この操作中はノードのネットワーク設定は保持されます。クラスタに参加しているノードは工場出荷時の設定にリセットされません。

パラメータ

ResetNode API は、「使用可能」状態のノードでのみ使用できます。クラスター内で「アクティブ」なノード、または「保留中」状態のノードでは使用できません。

警告：

このメソッドは、ノード上にあるすべての顧客データをクリアします。

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
build	ノードがリセットされるリモートElementソフトウェアイメージへのURLを指定するために使用されます。	URL	なし	いいえ
力	ノードをリセットするには true に設定します。	ブーリアン	なし	はい
options	リセット操作を実行するための仕様を入力するために使用されます。詳細が必要な場合は、NetAppサポートにお問い合わせください。	JSONオブジェクト	なし	いいえ

戻り値

このメソッドには戻り値はありません。

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "ResetNode",
  "params": {
    "build" : "file:///sf/rtfi/image/filesystem.squashfs",
    "force" : true
  },
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": null,
  "result": {
    "rtfiInfo": {
```

```

"build": "file:///sf/rtfi/image/filesystem.squashfs",
"generation": "9",
"options": {
  "edebug": "",
  "sf_auto": "0",
  "sf_bond_mode": "ActivePassive",
  "sf_check_hardware": "0",
  "sf_disable_otpw": "0",
  "sf_fa_host": "",
  "sf_hostname": "SF-FA18",
  "sf_inplace": "1",
  "sf_inplace_die_action": "kexec",
  "sf_inplace_safe": "0",
  "sf_keep_cluster_config": "0",
  "sf_keep_data": "0",
  "sf_keep_hostname": "0",
  "sf_keep_network_config": "0",
  "sf_keep_paths": "\"/var/log/hardware.xml\"",
  "sf_max_archives": "5",
  "sf_nvram_size": "",
  "sf_oldroot": "",
  "sf_postinst_erase_root_drive": "0",
  "sf_root_drive": "",
  "sf_rtfi_cleanup_state": "",
  "sf_secure_erase": "1",
  "sf_secure_erase_retries": "5",
  "sf_slice_size": "",
  "sf_ssh_key": "1",
  "sf_ssh_root": "1",
  "sf_start_rtfi": "1",
  "sf_status_httpserver": "1",
  "sf_status_httpserver_stop_delay": "5m",
  "sf_status_inject_failure": "",
  "sf_status_json": "0",
  "sf_support_host": "sfsupport.solidfire.com",
  "sf_test_hardware": "0",
  "sf_upgrade": "0",
  "sf_upgrade_firmware": "0",
  "sf_upload_logs_url": ""
},
"statusUrlAll": "http://192.168.130.20/status/all.json",
"statusUrlCurrent": "http://192.168.130.20/status/current.json"
}
}

```

リセットノード補足Tls暗号

使用することができます `ResetNodeSupplementalTlsCiphers` 補足 TLS 暗号のリストをデフォルトに復元する方法。このコマンドは管理ノードで使用できます。

パラメータ



このメソッドは管理ノードに対して呼び出す必要があります。例えば：

```
https://<management node IP>:442/json-rpc/10.0
```

このメソッドには入力パラメータはありません。

戻り値

このメソッドには戻り値はありません。

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "ResetNodeSupplementalTlsCiphers",
  "params": {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id" : 1,
  "result" : {}
}
```

再起動ネットワーク

使用することができます `RestartNetworking` ノード上のネットワーク サービスを再起動する方法。

警告：

この方法では、ノード上のすべてのネットワーク サービスが再起動されるため、ネットワーク接続が一時的に失われます。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
力	ノード上のネットワーク サービスを再起動するには true に設定します。	ブーリアン	なし	はい

戻り値

このメソッドには戻り値はありません。

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "RestartNetworking",
  "params": {
    "force" : true
  },
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{ "id" : 1,
  "result" : {}
}
```


サービスの再起動

使用することができます `RestartServices` ノード上のサービスを再起動する方法。

パラメータ

警告：

この方法では、一時的にノード サービスの中断が発生します。

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
力	ノード上のサービスを再起動するには true に設定します。	ブーリアン	なし	はい
service	再起動するサービス名。	string	なし	いいえ
アクション	サービスに対して実行するアクション (開始、停止、再起動)。	string	なし	いいえ

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
詳細	エラー (ある場合) を含む、サービス再起動手順の出力。	JSONオブジェクト
duration	ノードへのサービスを再開するのにかった時間 (秒) 。	string
結果	再起動の結果。	string

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "RestartServices",
  "params": {
    "force" : true
    "action" : restart,
  }
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "details": "solidfire stop/waiting\nsolidfire start/running, process
7284\n",
    "duration": "00:00:02.541594",
    "result": "Passed"
  }
}
```

バージョン以降の新機能

9.6

クラスタ構成の設定

使用することができます SetClusterConfig`ノードが関連付けられているクラスターと通信するために使用する構成を設定する方法。ノードの現在のクラスタインターフェース設定を表示するには、`GetClusterConfig API メソッド。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
cluster	このメソッド呼び出し中に変更する必要がある構成属性。変更するフィールドのみを、このパラメータのメンバーとしてこのメソッドに追加する必要があります。	cluster	なし	いいえ

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
cluster	ノードがクラスターと通信するために使用する構成情報。	cluster

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "SetClusterConfig",
  "params": {
    "cluster": {
      "name": "myhost",
      "mipi": "Bond10G"
    },
    "id" : 1
  }
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id" : 1,
  "result" : {
    "cluster" : {
      "cipi" : "Bond10G",
      "cluster" : "QoS",
      "ensemble" : [
        "1:10.10.5.42",
        "2:10.10.5.43",
        "3:10.10.5.44",
        "4:10.10.5.46",
        "5:10.10.5.47"
      ],
      "hostname" : "myhost",
      "mipi" : "Bond10G",
      "nodeID" : 1,
      "sipi" : "Bond10G",
      "state" : "Active"
    }
  }
}
```

バージョン以降の新機能

9.6

設定

使用することができます `SetConfig` ノードのネットワークとクラスター情報を設定するメソッド。このメソッドは、両方のAPIメソッドで利用できる同じ設定を単一のAPIメソッドに含めます。`SetClusterConfig` そして `SetNetworkConfig` 方法。このメソッドでは、変更するフィールドのみを含める必要があります。

パラメータ

警告：

ノードのボンド モードを変更すると、ネットワーク接続が一時的に失われる可能性があります。

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
cluster	ストレージ ノードが関連付けられているストレージ クラスターと通信する方法を識別するクラスター情報。	cluster	なし	いいえ
network	ノードの各ネットワーク インターフェイスのネットワーク 接続タイプと現在の設定。	ネットワーク（すべてのインターフェース）	なし	いいえ

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
設定	<p>ノードの新しい現在の構成。このオブジェクトには次のものが含まれます。</p> <ul style="list-style-type: none"> • cluster: ストレージ ノードが関連付けられているストレージ クラスターと通信する方法を識別するクラスター情報。 • ネットワーク（すべてのインターフェース）: ノードの各ネットワーク インターフェイスのネットワーク接続タイプと現在の設定。 	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "SetConfig",
  "params": {
    "cluster": {
      "name": "MyHostname"
    },
    "network": {
      "Bond10G": {
        "bond-mode": "ALB"
      }
    }
  }
}
```

応答例

このメソッドからの応答は、GetConfig メソッドの戻り値と同じです。SetConfig を使用すると、オブジェクトのすべてのフィールドが表示され、更新された値が表示されます。

バージョン以降の新機能

9.6

詳細情報の参照

- [クラスタ構成の設定](#)
- [ネットワーク設定の設定](#)
- [ゲットコンフィグ](#)

ネットワーク設定の設定

使用することができます SetNetworkConfig`ノードのネットワーク構成を設定する方法。ノードの現在のネットワーク設定を表示するには、`GetNetworkConfig API メソッド。

パラメータ

警告：

ノードのボンド モードを変更すると、ネットワーク接続が一時的に失われる可能性があります。

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
network	変更するノード ネットワーク設定を含むオブジェクト。このメソッドに変更するフィールドを、このパラメータの属性として追加するだけです。	ネットワーク（すべてのインターフェース）	なし	いいえ

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
network	ノードの新しい現在のネットワーク構成。	ネットワーク（すべてのインターフェース）

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "SetNetworkConfig",
  "params": {
    "network": {
      "Bond10G": {
        "bond-mode": "ALB"
      },
      "Bond1G": {
        "netmask": "255.255.224.0"
      },
      "eth0": {
        "method": "bond"
      },
      "lo": {
        "method": "loopback"
      }
    }
  }
}
```

応答例

このメソッドからの応答は、GetNetworkConfig メソッドからの応答と同じです。このメソッドは、各オブジェクトのすべてのメンバーを表示し、変更されたメンバーの新しい値を含めます。

バージョン以降の新機能

9.6

詳細情報の参照

- [ネットワーク構成の取得](#)
- [ネットワーク構成の取得](#)

ノードSSL証明書の設定

使用することができます `SetNodeSSLCertificate` 管理ノードのユーザー SSL 証明書と秘密キーを設定する方法。



API を使用した後は、管理ノードを再起動する必要があります。

パラメータ



このメソッドは管理ノードに対して呼び出す必要があります。例えば：

```
https://<management node IP>:442/json-rpc/10.0
```

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
証明書	証明書の PEM エンコードされたテキスト バージョン。注: ノードまたはクラスター証明書を設定する場合、証明書には serverAuth の extendedKeyUsage 拡張機能が含まれている必要があります。この拡張機能により、一般的なオペレーティング システムとブラウザーで証明書をエラーなく使用できるようになります。拡張機能が存在しない場合、API は証明書を無効として拒否します。	string	なし	はい
秘密鍵	秘密鍵の PEM エンコードされたテキスト バージョン。	string	なし	はい

戻り値

このメソッドには戻り値はありません。

リクエスト例

このメソッドのリクエストは次の例ようになります。

```
{
  "method" : "SetNodeSSLCertificate",
  "params" : {
    "privateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIIEowIBAAKCAQEA8U+28fnLKQNWEMMR6akeDKuehSpS79odLGigI18qlCV/AUY5\nZLjqsT
jBvTJVRv44yoCTgNrx36U7FHP4t6P/Si0aYr4ovxl5wDpEM3Qyy5JPB7Je\nlOB6AD7fmiTweP
20HRYpZvY+Uz7LYEFCmrpgGZQF3iOSIcBHtLKE5186JVT6j5dg\n6yjUGQO352ylc9HXHcn6lb
/jy10DmVNUZ0caQwAmIS3Jmoyx+zj/Ya4WKq+2SqTA\nX7bX0F3WHHfXnZ1HnM8fET5N/9A+K6
lS7dg9cyXu4afXcgKy14JiNBvqbBjhGJtE\n76yAy6rTHu0xM3jjdkcb9Y8miNzxF+ACq+itaw
IDAQABAOIBAH1jlIzr6/sltqVW\n00qVC/49dyNu+KWVSq92ti9rFe7hBPueh9gklh78hP9Qli
tLkir3YK4GFsTFUMux\n7z1NRCxA/4LrmLSkAjW2kRXDfVl2bwZq0ua9NefGw92O8D2OZvbuOx
k7Put2p6se\nfngNzSjf2SI5DIX3UME5dDN5FByu52CJ9mI4U16ngbWln2wc4nsxJg0aAEkzB7w
nq\nt+Am5/Vu1LI6rGiG6oHEW0oGSuH1lesIyXXa2hqqU+1+iF2iGRMTiXac4C8d11NU\nnWGIR
```



```
{
  "id" : 2,
  "result" : {}
}
```

ノード補足Tls暗号の設定

使用することができます `SetNodeSupplementalTlsCiphers` 補足 TLS 暗号のリストを指定する方法。このコマンドは管理ノードで使用できます。

パラメータ



このメソッドは管理ノードに対して呼び出す必要があります。例えば：

```
https://<management node IP>:442/json-rpc/10.0
```

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
補足暗号	OpenSSL 命名スキームを使用した補足暗号スイート名。暗号スイート名の使用では大文字と小文字は区別されません。	string	なし	はい

戻り値

このメソッドには次の戻り値があります。

Name	説明	タイプ
必須暗号	ノードの必須 TLS 暗号スイートのリスト。これらはノード上で常にアクティブな暗号です。	string
補足暗号	ノードの補足 TLS 暗号スイートのリスト。	string

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "SetNodeSupplementalTlsCiphers",
  "params": {
    "supplementalCiphers": [
      "DHE-RSA-AES128-SHA256",
      "DHE-RSA-AES128-GCM-SHA256",
      "ECDHE-RSA-AES128-SHA256",
      "ECDHE-RSA-AES128-GCM-SHA256"
    ]
  },
  "id": 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id" : 1,
  "result" : {
    "mandatoryCiphers": [
      "DHE-RSA-AES256-SHA256",
      "DHE-RSA-AES256-GCM-SHA384",
      "ECDHE-RSA-AES256-SHA384",
      "ECDHE-RSA-AES256-GCM-SHA384"
    ],
    "supplementalCiphers": [
      "DHE-RSA-AES128-SHA256",
      "DHE-RSA-AES128-GCM-SHA256",
      "ECDHE-RSA-AES128-SHA256",
      "ECDHE-RSA-AES128-GCM-SHA256"
    ]
  }
}
```

シャットダウン

使用することができます `Shutdown` クラスター内のノードを再起動またはシャットダウンする方法。この方法を使用すると、クラスター内の単一のノード、複数のノード、またはすべてのノードをシャットダウンできます。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
ノード	再起動またはシャットダウンするノードの NodeID のリスト。	整数配列	なし	はい
option	<p>クラスターに対して実行するアクション。有効な値は次のとおりです。</p> <ul style="list-style-type: none">• restart: クラスターを再起動します。• halt: 完全な電源オフを実行します。	string	restart	いいえ

戻り値

このメソッドには戻り値はありません。

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "Shutdown",
  "params": {
    "nodes": [
      2,
      3,
      4
    ],
    "option": "halt"
  },
  "id": 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id" : 1,
  "result" : {
    "failed": [],
    "successful": [
      6
    ]
  }
}
```

バージョン以降の新機能

9.6

テストコネクトアンサンブル

使用することができます `TestConnectEnsemble` 指定されたデータベース アンサンブルとの接続を確認する方法。デフォルトでは、ノードが関連付けられているクラスターのアンサンブルが使用されます。あるいは、接続性をテストするために別のアンサンブルを提供することもできます。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
アンサンブル	接続テスト用のアンサンブル ノード クラスター IP アドレスのコンマ区切りリスト。	string	なし	いいえ

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
詳細	返されるオブジェクト: <ul style="list-style-type: none"> • nodes: (オブジェクト) テスト内の各アンサンブル ノードとテストの結果のリスト。 • duration: (文字列) テストの実行に必要な時間。 • result: (文字列) テスト全体の結果。 	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "TestConnectEnsemble",
  "params": {},
  "id": 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "details": {
      "nodes": {
        "1:10.10.20.70": "Passed",
        "2:10.10.20.71": "Passed",
        "3:10.10.20.72": "Passed",
        "4:10.10.20.73": "Passed",
        "5:10.10.20.74": "Passed"
      }
    },
    "duration": "00:00:00:756072",
    "result": "Passed"
  }
}
```

テストコネクトMvip

使用することができます `TestConnectMvip` ストレージ クラスターへの管理接続をテストする方法。テストは MVIP に ping を送信し、接続を確認するための簡単な API メソッドを実行します。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
mvip	この値を渡すと、別の MVIP の管理接続をテストできます。ターゲット クラスターへの接続をテストするときは、この値を使用する必要はありません。	string	なし	いいえ

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
詳細	<p>テスト操作に関する情報 (JSON オブジェクト):</p> <ul style="list-style-type: none"> connected: テストがMVIPに接続できたかどうかを示します (ブール値) mvip: テスト対象のMVIP (文字列) pingBytes: 56バイトと1500バイトのpingテストの詳細 (オブジェクト) <ul style="list-style-type: none"> 56: 56 バイトの ping テストの結果 (JSON オブジェクト): <ul style="list-style-type: none"> individualResponseTimes: 各アンサンブルノードからの応答時間のリスト (文字列配列) individualStatus: 各アンサンブルノードからのpingステータスのリスト (ブール配列) responseTime: 平均ping応答時間 (文字列) successful: pingテストが成功したかどうかを示します (ブール値) 1500: 1500 バイトの ping テストの結果 (JSON オブジェクト): <ul style="list-style-type: none"> individualResponseTimes: 各アンサンブルノードからの応答時間のリスト (文字列配列) individualStatus: 各アンサンブルノードからのpingステータスのリスト (ブール配列) responseTime: 平均ping応答時間 (文字列) successful: pingテストが成功したかどうかを示します (ブール値) 	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例のようになります。duration: テストの実行に必要な時間の長さ（文字列）

```
{
  "method": "TestConnectMvip",
  "params": {
    "mvip" : "172.27.62.50"
  },
  "id":1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "details": {
      "connected": true,
      "mvip": "172.27.62.50",
      "pingBytes": {
        "1500": {
          "individualResponseTimes": [
            "00:00:00.000250",
            "00:00:00.000206",
            "00:00:00.000200",
            "00:00:00.000199",
            "00:00:00.000199"
          ],
          "individualStatus": [
            true,
            true,
            true,
            true,
            true
          ],
          "responseTime": "00:00:00.000211",
          "successful": true
        },
        "56": {
          "individualResponseTimes": [
            "00:00:00.000217",
            "00:00:00.000122",

```

```

        "00:00:00.000117",
        "00:00:00.000119",
        "00:00:00.000121"
    ],
    "individualStatus": [
        true,
        true,
        true,
        true,
        true
    ],
    "responseTime": "00:00:00.000139",
    "successful": true
}
}
},
"duration": "00:00:00.271244",
"result": "Passed"
}
}

```

バージョン以降の新機能

9.6

テスト接続Svip

使用することができます `TestConnectSvip` ストレージ クラスターへのストレージ接続をテストする方法。テストは ICMP パケットを使用して SVIP に ping し、成功すると iSCSI イニシエーターとして接続します。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
svip	この値を渡すと、別の SVIP の管理接続をテストできます。ターゲット クラスターへの接続をテストするときは、この値を使用する必要はありません。	string	なし	いいえ

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
詳細	<p>テスト操作に関する情報 (JSON オブジェクト):</p> <ul style="list-style-type: none"> • connected: テストがSVIPに接続できたかどうかを示します (ブール値) • svip: テスト対象のSVIP (文字列) • pingBytes: 56バイトと9000バイトのpingテストの詳細 (オブジェクト) <ul style="list-style-type: none"> ◦ 56: 56 バイトの ping テストの結果 (JSON オブジェクト): <ul style="list-style-type: none"> ▪ individualResponseTimes: 各アンサンブルノードからの応答時間のリスト (文字列配列) ▪ individualStatus: 各アンサンブルノードからのpingステータスのリスト (ブール配列) ▪ responseTime: 平均ping応答時間 (文字列) ▪ successful: pingテストが成功したかどうかを示します (ブール値) ◦ 9000: 9000 バイトの ping テストの結果 (JSON オブジェクト): <ul style="list-style-type: none"> ▪ individualResponseTimes: 各アンサンブルノードからの応答時間のリスト (文字列配列) ▪ individualStatus: 各アンサンブルノードからのpingステータスのリスト (ブール配列) ▪ responseTime: 平均ping応答時間 (文字列) ▪ successful: pingテストが成功したかどうかを示します (ブール値) 	string

リクエスト例

このメソッドのリクエストは次の例のようになります。

° duration: テストの実行に

```
{
  "method": "TestConnectSvip",
  "params": {
    "svip" : "172.27.62.50"
  },
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "details": {
      "connected": true,
      "pingBytes": {
        "56": {
          "individualResponseTimes": [
            "00:00:00.000152",
            "00:00:00.000132",
            "00:00:00.000119",
            "00:00:00.000114",
            "00:00:00.000112"
          ],
          "individualStatus": [
            true,
            true,
            true,
            true,
            true
          ],
          "responseTime": "00:00:00.000126",
          "successful": true
        },
        "9000": {
          "individualResponseTimes": [
            "00:00:00.000295",
            "00:00:00.000257",
            "00:00:00.000172",
```



```

        "00:00:00.000172",
        "00:00:00.000267"
    ],
    "individualStatus": [
        true,
        true,
        true,
        true,
        true
    ],
    "responseTime": "00:00:00.000233",
    "successful": true
    }
},
"svip": "172.27.62.50"
},
"duration": "00:00:00.421907",
"result": "Passed"
}
}

```

バージョン以降の新機能

9.6

テストドライブ

使用することができます `TestDrives` ノード上のすべてのドライブに対してハードウェア検証を実行する方法。この方法では、ドライブ上のハードウェア障害を検出し、検証テストの結果にそれを報告します。

パラメータ

使用できるのは `TestDrives` クラスター内で「アクティブ」ではないノード上のメソッド。



このテストには約 10 分かかります。

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
力	ノード上のドライブをテストするには true に設定します。	ブーリアン	なし	はい

Name	説明	タイプ	デフォルト値	必須
minutes	テストを実行する分 数を指定します。	integer	10	いいえ

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
詳細	テスト操作の成功または失敗に関する情報。	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "TestDrives",
  "params": {
    "force": true,
    "minutes" : 10
  },
  "id" : 1
}
```

応答例

このメソッドは、ノード内の各ドライブのテスト結果を含むテーブルを返します。

バージョン以降の新機能

9.6

テストハードウェア構成

使用することができます `TestHardwareConfig` ノード上でハードウェア テストを実行する方法。テスト オプションには、ハードウェア構成、ファームウェア バージョン、すべてのドライブが存在することの確認が含まれます。

パラメータ



これらのテストはハードウェア障害を検出することを目的としたものではありません。

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
clean	クリーンなキャッシュでハードウェア構成テストを開始します。有効な値は次のとおりです。 <ul style="list-style-type: none">• true: キャッシュされたテスト結果ファイルを削除し、テストを再実行します。• false: キャッシュされたテスト結果を取得します。	ブーリアン	false	いいえ
力	ノードを正常にリセットするには、このメソッドに force パラメータを含める必要があります。	ブーリアン	なし	はい

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
詳細	ハードウェア構成の詳細。	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "TestHardwareConfig",
  "params": {
    "force": true
  },
  "id" : 1
}
```

応答例

この応答例は長いため、補足トピックに記載されています。

バージョン以降の新機能

9.6

詳細情報の参照

[テストハードウェア構成](#)

テスト配置クラスター

使用することができます `TestLocateCluster` ノードがクラスター構成で指定されたクラスターを見つけられるかどうかを検証するメソッド。出力では、クラスターが作成されたことが検証され、クラスター アンサンブル内のノードが一覧表示されます。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
詳細	テスト操作の成功または失敗に関する情報。	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "TestLocateCluster",
  "params": {},
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": 1,
  "result": {
    "details": {
      "complete": true,
      "ensemble": {
        "nodes": [
          {
            "IP": "10.10.5.94",
            "nodeID": 1
          },
          {
            "IP": "10.10.5.107",
            "nodeID": 2
          },
          {
            "IP": "10.10.5.108",
            "nodeID": 3
          }
        ]
      },
      "version": "5.749"
    },
    "duration": "0.0384478sec",
    "result": "Passed"
  }
}
```

バージョン以降の新機能

9.6

ローカル接続のテスト

使用することができます `TestLocalConnectivity` アクティブ クラスター内の各ノードのクラスター IP (CIP) を ping する方法。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
詳細	ローカルのアクティブ クラスター内の各ノードの個別の ping 応答時間。	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例ようになります。

```
{
  "method": "TestLocalConnectivity",
  "params": {},
  "id": 1
}
```

応答例

このメソッドのリクエストは次の例ようになります。

```
{
  "id": null,
  "result": {
    "details": {
      "10.26.86.17": {
        individualResponseTimes: [
          "00:00:00.006868",
          "00:00:00.005933",
          "00:00:00.006655",
          "00:00:00.006584",
          "00:00:00.006334"
        ],
        individualStatus: [
          true,
          true,
          true,
          true,
          true
        ],
        responseTime: "00:00:00.006475",
        successful: true
      },
      "10.26.86.18": {
        individualResponseTimes: [
          "00:00:00.006201",
```

```

        "00:00:00.006187",
        "00:00:00.005990",
        "00:00:00.006029",
        "00:00:00.005917"],
    individualStatus: [
        true,
        true,
        true,
        true,
        true
    ],
    "responseTime": "00:00:00.006065",
    "successful": true
},

    "10.26.86.19": {
    individualResponseTimes: [
        "00:00:00.005988",
        "00:00:00.006948",
        "00:00:00.005981",
        "00:00:00.005964",
        "00:00:00.005942"
    ],
    individualStatus: [
        "true",
        "true",
        true,
        true,
        true
    ],
    responseTime: "00:00:00.006165",
    successful: true,
},

    "10.26.86.20": {
    individualResponseTimes: [
        "00:00:00.005926",
        "00:00:00.006072",
        "00:00:00.005675",
        "00:00:00.009904",
        "00:00:00.006225"
    ],
    "individualStatus": [
        true,
        true,
        true,
        true,
        true
    ]
}

```

```

        ],
        responseTime: "00:00:00.006760",
        successful: true
    }
},
"duration": "00:00:00.595982",
"result": "Passed"
}
}

```

バージョン以降の新機能

9.6

テストネットワーク構成

使用することができます `TestNetworkConfig` 構成されたネットワーク設定がシステムで使用されているネットワーク設定と一致しているかどうかをテストする方法。

パラメータ

UI または TUI で SetNetworkConfig メソッドを使用してノードを構成すると、構成が検証され、保存されます。TestNetworkConfig API テストでは、保存された構成を事後検証ロジックに使用します。たとえば、停電やネットワーク障害が発生した場合、この API メソッドを使用して、ノードが最新のネットワーク構成で実行されていることを確認できます。これにより、構成にエラーがないこと、および現在の構成が使用されていることが検証されます。

このテストは、応答出力の失敗のみを表示するように設計されています。エラーがない場合、このテストは出力を返しません。次の応答例を参照してください。

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
詳細	現在保存されているネットワーク設定を実行中のネットワーク構成で検証したときに見つかったエラーが含まれます。	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例のようになります。


```
{
  "method": "TestNetworkConfig",
  "params": {},
  "id" : 1
}
```

応答例1

エラーが検出されない場合、応答は返されません。

```
{
  "id" : 1,
  "result": {
    "details": {
      "network": {...}
    },
    "duration": "00:00:00.144514",
    "result": "Passed"
  }
}
```

応答例2

MTUが一致しない例。

```
{
  "id" : 1,
  "result":
  {
    "details" :
    {
      "error":
      {
        "message" : "Network configuration mismatch on Bond10G:
Incorrect MTU expectedMTU=[1500]  actualMTU=[9600]", name:
"xAssertionFailure"
      }
    },
    "duration": "0.125213sec",
    "result": "Failed"
  }
}
```

応答例3

静的ルートが見つからない場合の例。

```
{
  "id": 1,
  "result": {
    "details" : {
      "error": {
        "message" : "Network configuration mismatch on Bond1G: Routing
table missing route=[192.168.137.2 via 192.168.159.254 dev Bond1G]", name:
"xAssertionFailure"
      }
    },
    "duration" : "0.128547sec",
    "result" : "Failed"
  }
}
```

バージョン以降の新機能

9.6

詳細情報の参照

[ネットワーク設定の設定](#)

テストピング

使用することができます `TestPing` ICMP パケットを使用して、1G および 10G インターフェースの両方でクラスター内のすべてのノードへのネットワーク接続をテストする方法。テストでは、ネットワーク構成の MTU 設定に基づいて、各パケットに適切な MTU サイズが使用されます。`TestPing` 一時的な VLAN インターフェイスを作成しません。

パラメータ

このメソッドには次の入力パラメータがあります。

Name	説明	タイプ	デフォルト値	必須
試み	システムがテスト ping を繰り返す回数を指定します。	integer	5	いいえ
ホスト	ping するデバイスのアドレスまたはホスト名をコンマで区切ったリストを指定します。ホストが指定されていない場合、このメソッドはストレージ クラスター内のホストに ping を実行します。	string	なし	いいえ
インタフェース	ping を送信する既存の (ベース) インタフェース。有効な値は次のとおりです。 <ul style="list-style-type: none"> Bond10G: Bond10G インタフェースから ping を送信します。 Bond1G: Bond1G インタフェースから ping を送信します。 	string	なし	いいえ
パケットサイズ	各 IP に送信される ICMP パケットで送信するバイト数を指定します。バイト数は、ネットワーク構成で指定された最大 MTU 未満である必要があります。	integer	なし	いいえ
pingタイムアウトミリ秒	個々の ping 応答を待機するミリ秒数を指定します。	integer	500ミリ秒	いいえ
断片化を禁止する	ICMP パケットの DF (Do not Fragment) フラグを有効にします。	ブーリアン	false	いいえ

Name	説明	タイプ	デフォルト値	必須
ソースアドレスV4	ICMP ping パケットで使用する送信元 IPv4 アドレス。	string	なし	いいえ
ソースアドレスV6	ICMP ping パケットで使用する送信元 IPv6 アドレス。	string	なし	いいえ
合計タイムアウト秒	次の ping 試行を発行するかプロセスを終了する前に、ping がシステム応答を待機する時間を秒単位で指定します。	integer	5	いいえ
仮想ネットワークタグ	ping パケットを送信するときに使用する VLAN ID。	integer	なし	いいえ

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
詳細	ノードが通信できた各 IP のリストと ping 応答統計。	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "TestPing",
  "params": {
    "interface": "Bond1G",
    "hosts": "192.168.0.1"
  },
  "id" : 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```

{
  "id": 1,
  "result": {
    "details": {
      "192.168.0.1": {
        "individualResponseCodes": [
          "Success",
          "Success",
          "Success",
          "Success",
          "Success"
        ],
        "individualResponseTimes": [
          "00:00:00.000304",
          "00:00:00.000123",
          "00:00:00.000116",
          "00:00:00.000113",
          "00:00:00.000111"
        ],
        "individualStatus": [
          true,
          true,
          true,
          true,
          true
        ],
        "interface": "Bond1G",
        "responseTime": "00:00:00.000154",
        "sourceAddressV4": "192.168.0.5",
        "successful": true
      }
    },
    "duration": "00:00:00.001747",
    "result": "Passed"
  }
}

```

バージョン以降の新機能

5.0

リモート接続のテスト

使用できます `TestRemoteConnectivity` リモート クラスターの各ノードを ping し、リモ

ート アンサンブル データベース接続を確認する方法。このメソッドで有用な結果を返すには、クラスターをペアにする必要があります。リモート データベース接続が失敗した場合、システムからの応答に例外がリストされます。

パラメータ

このメソッドには入力パラメータはありません。

戻り値

このメソッドの戻り値は次のとおりです。

Name	説明	タイプ
詳細	各ノードの個別の ping 応答時間。	JSONオブジェクト

リクエスト例

このメソッドのリクエストは次の例のようになります。

```
{
  "method": "TestRemoteConnectivity",
  "params": {
    "force": "true"
  },
  "id": 1
}
```

応答例

このメソッドは、次の例のような応答を返します。

```
{
  "id": null,
  "result": {
    "details": {
      "1": {
        "details": {
          "10.26.86.17": {
            "individualResponseTimes": [
              "00:00:00.006868",
              "00:00:00.005933",
              "00:00:00.006655",
              "00:00:00.006584",
              "00:00:00.006334"
            ]
          }
        }
      }
    }
  }
}
```

```

    ],
    "individualStatus": [
        "true",
        "true",
        "true",
        "true",
        "true"
    ],
    "responseTime": "00:00:00.006475",
    "successful": true
},
"10.26.86.18": {
    "individualResponseTimes": [
        "00:00:00.006201",
        "00:00:00.006187",
        "00:00:00.005990",
        "00:00:00.006029",
        "00:00:00.005917"
    ],
    "individualStatus": [
        "true",
        "true",
        "true",
        "true",
        "true"
    ],
    "responseTime": "00:00:00.006065",
    "successful": true
},
"10.26.86.19": {
    "individualResponseTimes": [
        "00:00:00.005988",
        "00:00:00.006948",
        "00:00:00.005981",
        "00:00:00.005964",
        "00:00:00.005942"
    ],
    "individualStatus": [
        "true",
        "true",
        "true",
        "true",
        "true"
    ],
    "responseTime": "00:00:00.006165",
    "successful": true,

```

```

    },
    "10.26.86.20": {
      "individualResponseTimes": [
        "00:00:00.005926",
        "00:00:00.006072",
        "00:00:00.005675",
        "00:00:00.009904",
        "00:00:00.006225"
      ],
      "individualStatus": [
        "true",
        "true",
        "true",
        "true",
        "true"
      ],
      "responseTime": "00:00:00.006760",
      "successful": true
    }
  },
  "successful": true
}
},
"duration": "00:00:00.595982",
"result": "Passed"
}
}

```

バージョン以降の新機能

9.6

著作権に関する情報

Copyright © 2025 NetApp, Inc. All Rights Reserved. Printed in the U.S. このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

ネットアップの著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、ネットアップによって「現状のまま」提供されています。ネットアップは明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。ネットアップは、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

ネットアップは、ここに記載されているすべての製品に対する変更を随時、予告なく行う権利を保有します。ネットアップによる明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、ネットアップは責任を負いません。この製品の使用または購入は、ネットアップの特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許によって保護されている場合があります。

権利の制限について：政府による使用、複製、開示は、DFARS 252.227-7013（2014年2月）およびFAR 5252.227-19（2007年12月）のRights in Technical Data -Noncommercial Items（技術データ - 非商用品目に関する諸権利）条項の(b)(3)項、に規定された制限が適用されます。

本書に含まれるデータは商用製品および / または商用サービス（FAR 2.101の定義に基づく）に関係し、データの所有権はNetApp, Inc.にあります。本契約に基づき提供されるすべてのネットアップの技術データおよびコンピュータ ソフトウェアは、商用目的であり、私費のみで開発されたものです。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用权を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc.の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用权については、DFARS 252.227-7015(b)項（2014年2月）で定められた権利のみが認められます。

商標に関する情報

NetApp、NetAppのロゴ、<http://www.netapp.com/TM>に記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。