

株式会社エクストランス

X-MON3

X-MON クラスタ管理リファレンス

2018/7/18 版

まえがき

本書は X-MON クラスタ版を管理するリファレンスとなっております。

そのため、基本的な OS や CUI、GUI の一般的な操作、用語などについては知識をご理解の上でお読みください。

また、X-MON の操作画面はお使いの OS やブラウザ、ターミナルソフトによって異なる場合がございます。

本書で使用している IP アドレスは実際と環境と読み替えてご確認ください。

X-MON 自体の監視方法、使用方法については別途マニュアルもしくはオンラインヘルプをご参照ください。

- ・ 本書における解説環境

X-MON ver 3.0.5

本書以外のマニュアルについては X-MON サポートページにログインしてご確認ください。

<http://x-mon.jp/support/>

2012 年 11 月

改定履歴		
2012 年 11 月	初版	
2013 年 01 月	二版	
2013 年 02 月	三版	
2016 年 12 月	四版	
2017 年 10 月	五版	
2018 年 07 月	六版	

Copyright © 2004-2013 X-TRANS, Inc. All Rights Reserved.

目次

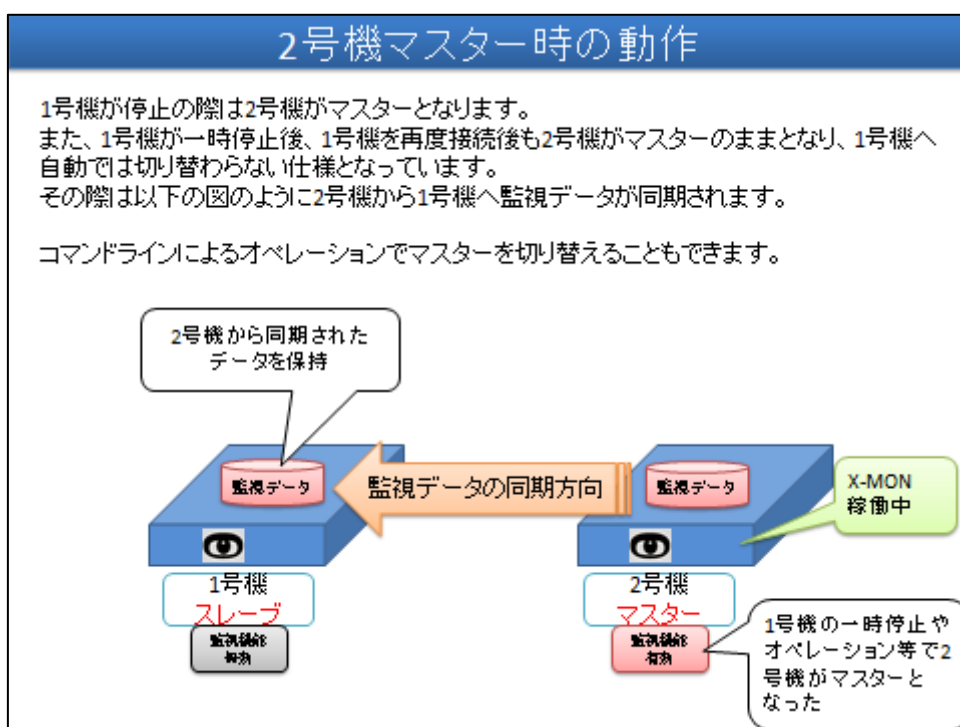
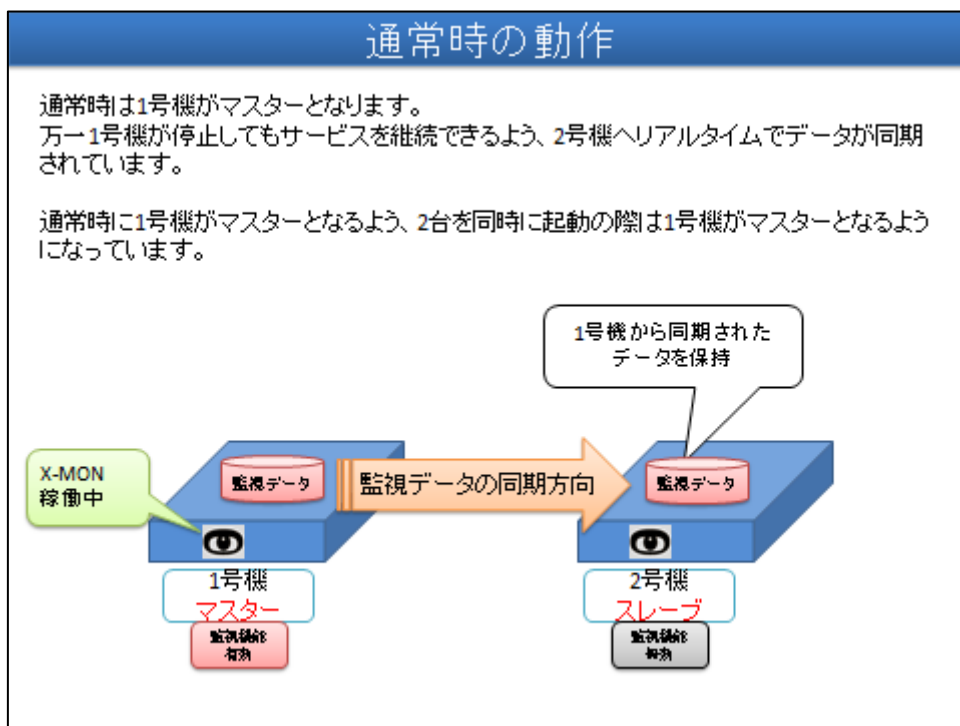
1	X-MON クラスタの仕様について	4
1.1	クラスタを形成するソフトウェア	5
1.2	用語について	6
1.3	本リファレンスの記載例	6
2	X-MON クラスタでの確認方法	7
2.1	クラスタのマスターとスレーブを確認する	7
2.1.1	デフォルトの構成の場合	7
2.1.2	切り替えを実施している場合	10
2.2	データ領域のプライマリとセカンダリを確認する	12
2.2.1	デフォルトの構成の場合	12
2.2.2	切り替えを実施している場合	14
2.2.3	障害が発生している場合	15
2.3	IP アドレスを確認する	17
2.3.1	デフォルトの構成の場合	17
2.3.2	切り替えを実施している場合	17
2.4	データ領域のマウントを確認する	18
2.4.1	デフォルトの構成の場合	18
2.4.2	切り替えを実施している場合	19
3	クラスタのマスターを切り替える	20
3.1	切り替えコマンドの発行	20
3.2	切り替え後の確認 (xmon-cl1 スレーブ)	20
3.3	切り替え後の確認 (xmon-cl2 マスター)	21
3.4	X-MON にて確認	23
4	切り替えたクラスタのマスターをデフォルトに戻す	24
4.1	切り替えコマンドの発行	24
4.2	切り替え後の確認 (xmon-cl1 マスター)	25
4.3	切り替え後の確認 (xmon-cl2 スレーブ)	26
4.4	X-MON で確認	28
5	データ同期が外れた場合の復旧について	29
5.1	マスター、スレーブともに StandAlone 状態	29
5.2	マスターが StandAlone、スレーブが WFconnection 状態	31
6	機器の障害について	33
6.1	HARDWARE 監視	33
6.2	RAID 監視	34
7	OS の設定変更方法	35

7.1	IP アドレス変更手順	35
7.1.1	X-MON のマスターの確認	35
7.1.2	サービス用 IP アドレスの変更手順	35
7.1.3	同期用 IP アドレスの変更手順	42
7.2	snmptrapd のコミュニティ名の変更手順	53
7.2.1	X-MON のマスターの確認	53
7.2.2	変更手順	53

1 X-MON クラスタの仕様について

X-MON クラスタについてご説明させていただきます。

デフォルトは1号機がマスターとなります。

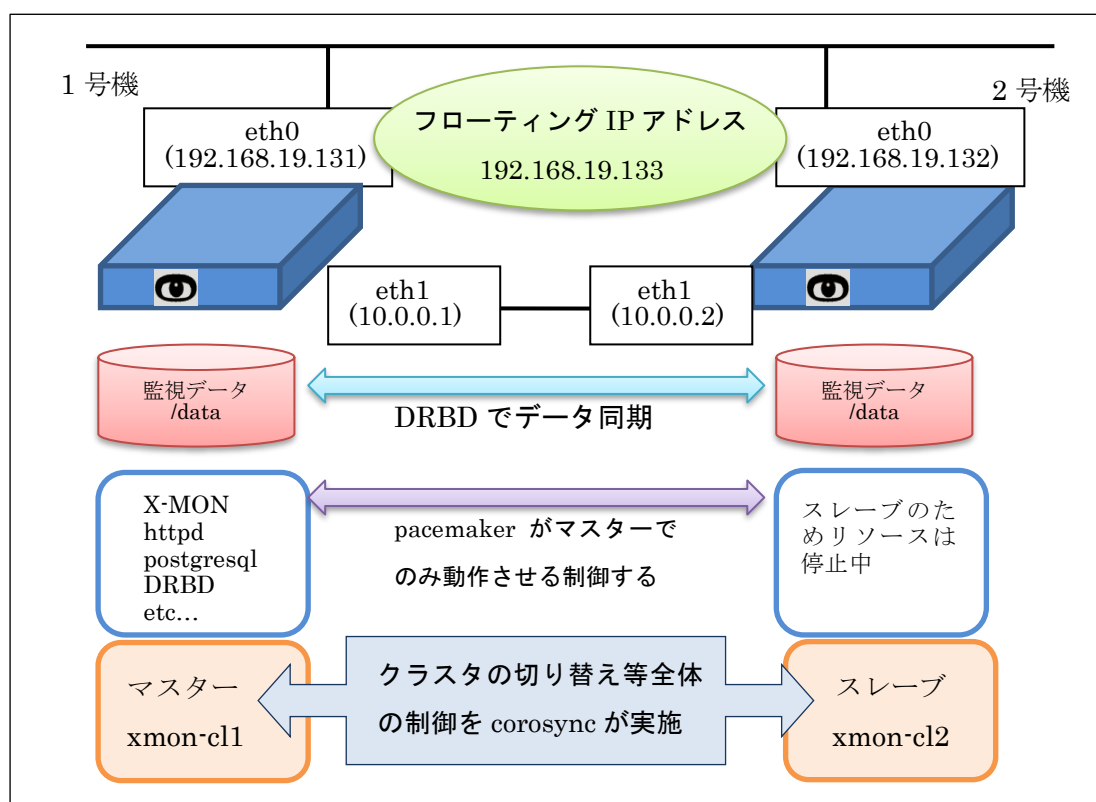


1.1 クラスタを形成するソフトウェア

X-MON クラスタを形成するソフトウェアについてご説明します。運用の際に重要となってきます。

DRBD	データ領域の同期を制御します。
pacemaker	クラスタのリソースと言われるフローティング IP アドレスや各プロセスを制御します。
corosync	クラスタ状態を制御し、クラスタの切り替えも制御します。

ソフトウェアの動作について、本リファレンスで使用するサンプルネットワークと共に下記に記載します。IP アドレスは実際と環境と読み替えてご確認ください



クラスタの状態を確認する際に、リソースとプロセスがマスターで動作しているか確認します。その際の一覧が下記となります。

プロセス名	クラスタでの設定名
x-mon	res_x-mon
ln-x-mon	res_ln-x-mon
snmptt	res_snmptt
nsca	res_nsca
httpd	res_httpd

postgresql	res_postgresql
drbd	res_drbd
snmptrapd	res_snmptrapd
リソースの設定	クラスタでの設定名
フローティング IP アドレス の制御	res_vip
データ同期する領域の制御	res_filesystem

1.2 用語について

■ クラスタ ・ マスター/スレーブ

マスターが X-MON を動作させているホスト、スレーブは X-MON を停止しているがデータの同期をマスターから受けており、いつでも稼働出来るようにしています。

■ データ領域 ・ プライマリ/セカンダリ

プライマリが X-MON のマスターでのデータ領域となり、スレーブ側がセカンダリとなります、データの同期はプライマリからセカンダリに対しての方向のみで実施します。

1.3 本リファレンスの記載例

■ 構文

<コマンド>

コマンドの構文です。

```
[root@xmon-cl1~ ] # <コマンド>
```

xmon-cl1 でのコマンド発行です。

```
[root@xmon-cl2 ~] # <コマンド>
```

xmon-cl2 でのコマンド発行です。

```
[root@xmon-cl1~ ] # <コマンド>
```

～中略～

コマンドを発行した際に、表示が長くなる部分は「～中略～」と記載しています。

```
hogehoge
```

コマンドを発行した結果や設定ファイルの変更箇所の解説の際に使用しています。

2 X-MON クラスタでの確認方法

2.1 クラスタのマスターとスレーブを確認する

クラスタのマスターとスレーブを確認するには `crm_mon` という管理コマンドを使用します。オプションで「-n1」を付ける事でノード毎に表示が出来ます。
本リファレンスでは、クラスタのマスターとプライマリを確認する際は

■ 構文

```
# crm_mon -n1
```

を使用します。

正常な場合、マスターとプライマリで発行した内容は同じ内容となります。

2.1.1 デフォルトの構成の場合

デフォルトのマスターである `xmon-cl1` で発行してみます。

```
[root@xmon-cl1 ~]# crm_mon -n1
=====
Last updated: Wed Nov 21 18:46:06 2012
Last change: Sun Nov 18 08:10:39 2012 via cibadmin on xmon-cl1
Stack: openais
Current DC: xmon-cl1 - partition with quorum
Version: 1.1.7-6.el6-148fccfd5985c5590cc601123c6c16e966b85d14
2 Nodes configured, 3 expected votes
11 Resources configured.
=====
Node xmon-cl1: online
    res_ln-x-mon    (lsb:ln-x-mon) Started
    res_snmpd       (lsb:snmpd) Started
    res_nsca        (lsb:nsca) Started
    res_vip         (ocf::heartbeat:IPAddr2) Started
    res_filesystem  (ocf::heartbeat:Filesystem) Started
    res_httpd       (lsb:httpd) Started
    res_postgresql  (lsb:postgresql) Started
    res_drbd:0      (ocf::linbit:drbd) Master
    res_x-mon       (lsb:x-mon) Started
    res_snmptrapd   (lsb:snmptrapd) Started
Node xmon-cl2: online
    res_drbd:1      (ocf::linbit:drbd) Started
```

クラスタのマスターのホスト名が表示されます。

クラスタに所属しているホストが online である事が表示され、そのホストで動作しているリソースが表示されます。

詳細についてご説明します。

```
Current DC: xmon-cl1 - partition with quorum
```

この行ではクラスタのマスターのホスト名が表示されます。

「xmon-cl1」と表示されていますので、マスターは xmon-cl1 である事がわかります。

```
Node xmon-cl1: online
```

```
res_ln-x-mon    (lsb:ln-x-mon) Started
res_snmpdt      (lsb:snmpdt) Started
res_nsca        (lsb:nsca) Started
res_vip (ocf::heartbeat:IPaddr2) Started
res_filesystem  (ocf::heartbeat:Filesystem) Started
res_httpd       (lsb:httpd) Started
res_postgresql  (lsb:postgresql) Started
res_drbd:0      (ocf::linbit:drbd) Master
res_x-mon       (lsb:x-mon) Started
res_snmptrapd    (lsb:snmptrapd) Started
```

クラスタに所属しているホストの状態を表示します、「online」は正常な状態です。停止していたりしてクラスタから外れている場合は「offline」となります。

また、そのホストで稼働しているリソースを表示します。この場合マスターである xmon-cl1 にてすべてのリソースが稼働している事がわかります。

```
Node xmon-cl2: online
```

```
res_drbd:1      (ocf::linbit:drbd) Started
```

スレーブである xmon-cl2 では、データ領域の制御を行う DRBD のみ稼働します。

DRBD のプライマリ・セカンダリの確認方法については [2.2 データ領域のプライマリとセカンダリを確認する](#) をご参照ください。

デフォルトのスレーブである xmon-cl2 で発行してみます。

```
[root@xmon-cl2 ~]# crm_mon -n1
```

```
=====
```

```
Last updated: Wed Nov 21 18:56:28 2012
```

```
Last change: Sun Nov 18 08:10:39 2012 via cibadmin on xmon-cl1
```

```
Stack: openais
```

```
Current DC: xmon-cl1 - partition with quorum
```

```
Version: 1.1.7-6.el6-148fccfd5985c5590cc601123c6c16e966b85d14
```

```
2 Nodes configured, 3 expected votes
```

```
11 Resources configured.
```

```
Node xmon-cl1: online
```

```
res_ln-x-mon    (lsb:ln-x-mon) Started
res_snmpptt     (lsb:snmpptt) Started
res_nsca        (lsb:nsca) Started
res_vip (ocf::heartbeat:IPaddr2) Started
res_filesystem  (ocf::heartbeat:Filesystem) Started
res_httpd       (lsb:httpd) Started
res_postgresql  (lsb:postgresql) Started
res_drbd:0      (ocf::linbit:drbd) Master
res_x-mon       (lsb:x-mon) Started
res_snmptrapd    (lsb:snmptrapd) Started
```

```
Node xmon-cl2: online
```

```
res_drbd:1      (ocf::linbit:drbd) Started
```

マスターである xmon-cl1 と同じ結果が表示されました。

正常なクラスタであれば、同じ結果が表示されます。結果が表示されない、違う内容の場合は故障の可能性があります。

マスターがどちらのホストであるか、X-MON のサービス一覧表示からも確認が出来ます。ホスト ID「X-MON」が X-MON 自身の監視となっています。サービス ID がマスターで動いているホストがどれかをステータス情報に表示します。

図 X-MON 監視

X-MON (X-MON)		Master Node	正常(OK)	2012-11-21 16:12:17	9日と09時間07分33秒	1/3	Master node: xmon-cl1
		PING	正常(OK)	2012-11-21 16:08:59	9日と09時間06分04秒	1/3	PING OK - Packet loss = 0%, RTA = 0.03 ms

図 ステータス情報

```
Master node: xmon-cl1
```

Master node:xmon-cl1 と表示されています。

これでマスターが動作しているサーバは xmon-cl1 だと確認できます。


```
Node xmon-cl2: online
    res_snmpdt (lsb:snmpdt) Started
    res_ln-x-mon (lsb:ln-x-mon) Started
    res_nsca (lsb:nsca) Started
    res_vip (ocf::heartbeat:IPAddr2) Started
    res_filesystem (ocf::heartbeat:Filesystem) Started
    res_httpd (lsb:httpd) Started
    res_postgresql (lsb:postgresql) Started
    res_drbd:1 (ocf::linbit:drbd) Master
    res_x-mon (lsb:x-mon) Started
    res_snmptrapd (lsb:snmptrapd) Started
```

マスターになった xmon-cl2 で各リソースが動作しているのが確認できます。

マスターに切り替わっている xmon-cl2 で発行してみます。

```
[root@xmon-cl2 ~]# crm_mon -n1
=====
Last updated: Wed Nov 21 16:22:16 2012
Last change: Sun Nov 18 08:10:39 2012 via cibadmin on xmon-cl1
Stack: openais
Current DC: xmon-cl2 - partition with quorum
Version: 1.1.7-6.el6-148fccfd5985c5590cc601123c6c16e966b85d14
2 Nodes configured, 3 expected votes
11 Resources configured.
=====

Node xmon-cl1: online
    res_drbd:0 (ocf::linbit:drbd) Started
Node xmon-cl2: online
    res_snmpdt (lsb:snmpdt) Started
    res_ln-x-mon (lsb:ln-x-mon) Started
    res_nsca (lsb:nsca) Started
    res_vip (ocf::heartbeat:IPAddr2) Started
    res_filesystem (ocf::heartbeat:Filesystem) Started
    res_httpd (lsb:httpd) Started
    res_postgresql (lsb:postgresql) Started
    res_drbd:1 (ocf::linbit:drbd) Master
```

```
res_x-mon      (lsb:x-mon) Started
res_snmptrapd  (lsb:snmptrapd) Started
```

xmon-cl1 と同じ内容が表示され、正常にクラスタが動作している事が確認出来ました。

それでは X-MON の監視画面でも確認してみましょう。

X-MON の Master_Node 監視のステータス情報が「Master node:xmon-cl2」に変わっています。

図 cl2 がマスターの場合

X-MON (X-MON)		Master_Node	正常(OК)	2012-11-21 16:19:17	9日と09時間13分43秒	1/3	Master node: xmon-cl2
	PING		正常(OК)	2012-11-21 16:18:59	9日と09時間12分14秒	1/3	PING OK = Packet loss = 0%, RTA = 0.02 ms

図 cl2 がマスターの場合のステータス情報

Master node: xmon-cl2

マスターで動作しているのが xmon-cl2 だと確認できます。

2.2 データ領域のプライマリとセカンダリを確認する

データ領域のプライマリとセカンダリを確認するには、データ領域を制御している DRBD のステータスを確認するコマンドとなります。

■ 構文

```
# service drbd status
```

2.2.1 デフォルトの構成の場合

デフォルトのマスターである xmon-cl1 で発行してみます。

```
[root@xmon-cl1 ~]# service drbd status
drbd driver loaded OK; device status:
version: 8.4.2 (api:1/proto:86-101)
GIT-hash: 7ad5f850d711223713d6dcadc3dd48860321070c build by dag@Build64R6,
2012-09-06 08:16:10
```

```
m:res cs ro ds p mounted fstype
00:r0 Connected Primary/Secondary UpToDate/UpToDate C /data ext4
```

同期する相手との
接続状況を表示

コマンドを発行したホスト
から見たデータ領域の状態
を表示

ディスクの状態
を表示

マウントポイントとファ
イルシステムを表示

cs
Connected

同期する相手と DRBD の接続状況を表示します。
Connected で正常に接続できています。

ro
Primary/Secondary

発行したホストから見たプライマリとセカンダリの状態を表示します。左側が自分のホスト上の状態となります。
そのため、xmon-cl1 はプライマリで動作している事がわかります。

ds
UpToDate/UpToDate

発行したホストから見たハードディスクの状態です。データの状態であるとも言えます。UpToDate が最新の状態であることを意味しています。プライマリ、セカンダリともに最新の状態である事が確認できます。

p mounted fstype
C /data ext4

データ領域の状態です。p はデータ同期を制御するプロトコルを表示しています。C で正常です。
mounted はデータ領域を/data でマウントし、fstype は ext4 のファイルシステムを使用している事を表示しています。

デフォルトのスレーブである xmon-cl2 で発行してみます。

```
[root@xmon-cl2 ~]# service drbd status
drbd driver loaded OK; device status:
version: 8.4.2 (api:1/proto:86-101)
GIT-hash: 7ad5f850d711223713d6dcadc3dd48860321070c build by dag@Build64R6,
2012-09-06 08:16:10
m:res cs ro ds p mounted fstype
0:r0 Connected Secondary/Primary UpToDate/UpToDate C
```

ro
Secondary/Primary

左側が自分のホスト上の状態となりますので、xmon-cl2 はセカンダリで動作している事がわかります。

p mounted fstype
C

データ領域の状態です。p はデータ同期を制御するプロトコルを表示しています。C で正常です。
セカンダリで動作していますので、データ制御のプロトコルは正常な C が表示されますがマウントはされないため空白となります。

X-MON でも正常にデータ領域が稼働しているか確認出来ます。

それぞれのホストで「DRBD_DISK」という監視が設定されています。

図 X-MON 監視

xmon-cl1 (xmon-cl1)	DRBD_DISK	正常(OK)	2012-11-21 20:03:59	0日と02時間41分17秒	1/3	DRBD OK: Device 0 Connected UpToDate
	PING	正常(OK)	2012-11-21 20:04:21	9日と07時間52分37秒	1/3	PING OK - Packet loss = 0%, RTA = 0.03 ms
xmon-cl2 (xmon-cl2)	DRBD_DISK	正常(OK)	2012-11-21 20:04:31	0日と02時間41分49秒	1/3	DRBD OK: Device 0 Connected UpToDate
	PING	正常(OK)	2012-11-21 20:04:27	9日と13時間08分43秒	1/3	PING OK - Packet loss = 0%, RTA = 0.19 ms

図 ステータス情報

DRBD OK: Device 0 Connected UpToDate

表示される内容は両ホスト共通です。

ステータス情報が「DRBD OK:Device 0 Connected UpToDate」が正常な状態となります。

2.2.2 切り替えを実施している場合

マスターの切り替えを実施して xmon-cl1 がスレーブ、xmon-cl2 がマスターの場合を確認してみましょう。

スレーブに切り替わっている xmon-cl1 で発行してみます。

```
[root@xmon-cl1 ~]# service drbd status
drbd driver loaded OK; device status:
version: 8.4.2 (api:1/proto:86-101)
GIT-hash: 7ad5f850d711223713d6dcadc3dd48860321070c build by dag@Build64R6,
2012-09-06 08:16:10
```

```
m:res cs ro ds p mounted fstype
0:r0 Connected Secondary/Primary UpToDate/UpToDate C
```

xmon-cl1 から見て「Secondary/Primary」になっているのでセカンダリで動作していて、なおかつマウントされていない事がわかります。

マスターに切り替わっている xmon-cl2 で発行してみます。

```
[root@xmon-cl2 ~]# service drbd status
drbd driver loaded OK; device status:
version: 8.4.2 (api:1/proto:86-101)
GIT-hash: 7ad5f850d711223713d6dcadc3dd48860321070c build by dag@Build64R6,
2012-09-06 08:16:10
```

```
m:res cs ro ds p mounted fstype
0:r0 Connected Primary/Secondary UpToDate/UpToDate C /data ext4
```

xmon-cl2 から見て「Primary/Secondary」になっているのでプライマリで動作していて、なおかつ/data を ext4 でマウントしている事がわかります。

2.2.3 障害が発生している場合

DRBD でデータの同期が出来なくなった場合を見てみましょう・

デフォルトのマスターの xmon-cl1 を確認してみます。

```
[root@xmon-cl1 ~]# service drbd status
drbd driver loaded OK; device status:
version: 8.4.2 (api:1/proto:86-101)
GIT-hash: 7ad5f850d711223713d6dcadc3dd48860321070c build by dag@Build64R6,
2012-09-06 08:16:10
m:res cs ro ds p mounted fstype
0:r0 StandAlone Primary/Unknown UpToDate/DUnknown r----- ext4
```

cs
StandAlone

同期する相手と DRBD の接続が出来ていない状態になると StandAlone（孤立している）の表示となります。

ro
Primary/Unknown

左側の自分の状態はプライマリである事が確認出来ますが、DRBD 接続が出来ていないので相手の状態がわからず Unknown と表示されます。

ds
UpToDate/DUnknown

左側の自分の状態は最新の状態である事が確認出来ますが、DRBD 接続が出来ていないので、相手の状態がわからず DUnknown と表示されます。

p mounted fstype
r----- ext4

障害が発生しているため、プロトコルの状態を表示する p は r の表示となり、マウントの状態は表示されなくなります。自分がプライマリ状態の場合は表示されませんがマウントされています。

X-MON でも障害を検知して確認出来ます。

図 障害発生時



WARNING を検知し、ステータス情報が両ホストとも共通で

「DRBD WARNING:Device 0 StandAlone UpToDate」となります。

スレーブである xmon-cl2 でも確認してみましょう。

```
[root@xmon-cl2 ~]# service drbd status
drbd driver loaded OK; device status:
GIT-hash: 7ad5f850d711223713d6dcadc3dd48860321070c build by dag@Build64R6,
2012-09-06 08:16:10
m:res cs ro ds p mounted fstype
0:r0 StandAlone Secondary/Unknown UpToDate/DUnknown r-----
```

cs
StandAlone

同期する相手と DRBD の接続が出来ていない状態
になると StandAlone の表示となります。

ro
Secondary/Unknown

左側の自分の状態はセカンダリである事が確認出来ますが、
DRBD 接続が出来ていないので相手の状態がわからず
Unknown と表示されます。

ds
UpToDate/DUnknown

左側の自分の状態は最新の状態である事が確認出来ますが、
DRBD 接続が出来ていないので、相手の状態がわからず
DUnknown と表示されます。

p mounted fstype
r-----

障害が発生しているため、プロトコルの状態を表示する p は
r の表示となり、マウントの状態は表示されなくなります。

スレーブである場合はもう一つのパターンがあります。

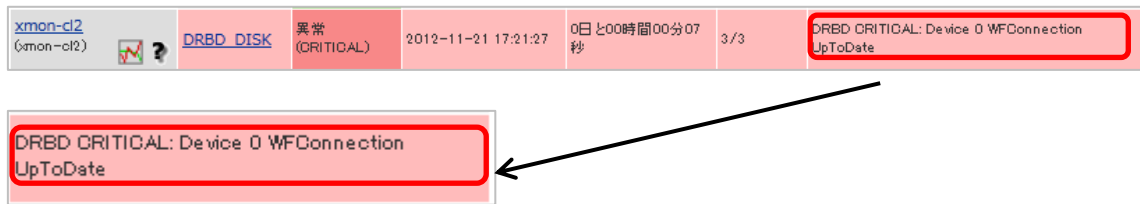
```
[root@xmon-cl2 ~]# service drbd status
drbd driver loaded OK; device status:
version: 8.4.2 (api:1/proto:86-101)
GIT-hash: 7ad5f850d711223713d6dcadc3dd48860321070c build by dag@Build64R6,
2012-09-06 08:16:10
m:res cs ro ds p mounted fstype
0:r0 WfConnection Secondary/Unknown UpToDate/DUnknown C
```

cs
Wfconnection

同期する相手と DRBD の接続を待っている状態の
場合は Wfconnection が表示されます。

X-MON でも障害を検知して確認出来ます。

図 障害発生時



Wfconnection の状態は接続待ち状態で CRITICAL を検知します。またこの障害は発生している時はスレーブホストのみがこの状態となります。

障害発生に対する復旧手順は [5 データ同期が外れた場合の復旧について](#) で解説します。

2.3 IP アドレスを確認する

マスターにはフローティング IP アドレスが割り当てられますので確認してみましょう。インタフェースは eth0:0 です。

■構文

```
# /sbin/ifconfig eth0:0
```

2.3.1 デフォルトの構成の場合

デフォルトのマスターである xmon-cl1 で発行してみましょう。

```
[root@xmon-cl1 ~]# /sbin/ifconfig eth0:0
eth0:0  Link encap:Ethernet  HWaddr 00:0C:29:AB:3A:25
        inet addr:192.168.19.133 Bcast:192.168.19.255 Mask:255.255.255.0
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

フローティング IP アドレスが割り当てられているのが確認出来ました。

デフォルトのスレーブである xmon-cl2 で発行してみます。

```
[root@xmon-cl2 ~]# /sbin/ifconfig eth0:0
eth0:0  Link encap:Ethernet  HWaddr 00:0C:29:0F:B3:D9
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

割り当てがありませんので、IP アドレスは表示されません。

2.3.2 切り替えを実施している場合

マスターの切り替えを実施して xmon-cl1 がスレーブ、xmon-cl2 がマスターの場合を確認してみましょう。

スレーブに切り替わっている xmon-cl1 で発行してみます。

```
[root@xmon-cl1 ~]# /sbin/ifconfig eth0:0
eth0:0    Link encap:Ethernet  HWaddr 00:0C:29:AB:3A:25
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

割り当てがありませんので、IP アドレスは表示されません。

マスターである xmon-cl2 で発行してみます。

```
[root@xmon-cl2 ~]# /sbin/ifconfig eth0:0
eth0:0    Link encap:Ethernet  HWaddr 00:0C:29:0F:B3:D9
          inet addr:192.168.19.133  Bcast:192.168.19.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

フローティング IP アドレスが割り当てられているのが確認出来ました。

IP アドレスが両ホストともに割り当てがされていない場合は障害が発生している可能性があります。

2.4 データ領域のマウントを確認する

マスターとスレーブで同期されているデータ領域が正常にマウントされているか確認します。

■ 構文

```
# df -Th
```

2.4.1 デフォルトの構成の場合

デフォルトのマスターである xmon-cl1 で発行してみましょう。

```
[root@xmon-cl1 ~]# df -Th
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/sda5       ext4      9.7G  901M  8.3G  10% /
tmpfs           tmpfs     3.9G   17M  3.9G   1% /dev/shm
/dev/sda1       ext4      485M   49M  412M  11% /boot
/dev/sda6       ext4      9.7G  1.1G  8.2G  12% /usr
/dev/sda2       ext4      20G   348M   18G   2% /var
/dev/drbd0      ext4     398G  1.1G  377G   1% /data
```

データ領域が ext4 ファイルシステムで/data でマウントされている事が確認出来ました。デフォルトのスレーブである xmon-cl2 で発行してみます。

```
[root@xmon-cl2 ~]# df -Th
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/sda5       ext4      9.7G  529M  8.7G   6% /
tmpfs           tmpfs     3.9G   17M  3.9G   1% /dev/shm
/dev/sda1       ext4      485M   49M  412M  11% /boot
/dev/sda6       ext4      9.7G  1.1G  8.2G  12% /usr
```

```
/dev/sda2    ext4    20G  333M  18G   2% /var
```

マウントされていないので、/data は表示されません。

2.4.2 切り替えを実施している場合

マスターの切り替えを実施して xmon-cl1 がスレーブ、xmon-cl2 がマスターの場合を確認してみましょう。

スレーブに切り替わっている xmon-cl1 で発行してみます。

```
[root@xmon-cl1 ~]# df -Th
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/sda5       ext4      9.7G  901M  8.3G  10% /
tmpfs           tmpfs      3.9G   17M  3.9G   1% /dev/shm
/dev/sda1       ext4      485M   49M  412M  11% /boot
/dev/sda6       ext4      9.7G  1.1G  8.2G  12% /usr
/dev/sda2       ext4      20G   348M   18G   2% /var
```

マウントされていないので、/data は表示されません。

マスターである xmon-cl2 で発行してみます。

```
[root@xmon-cl2 ~]# df -Th
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/sda5       ext4      9.7G  529M  8.7G   6% /
tmpfs           tmpfs      3.9G   17M  3.9G   1% /dev/shm
/dev/sda1       ext4      485M   49M  412M  11% /boot
/dev/sda6       ext4      9.7G  1.1G  8.2G  12% /usr
/dev/sda2       ext4      20G   333M   18G   2% /var
/dev/drbd0      ext4     398G  1.1G  377G   1% /data
```

マウントされている事が確認出来ました。

両ホストでマウントされている、両ホストでマウントされていない状態の場合は障害が発生している可能性があります。

3 クラスタのマスターを切り替える

デフォルトでは xmon-cl1 がマスター、xmon-cl2 がスレーブで動作しています。
xmon-cl1 が故障などで稼働を停止する必要がある場合、xmon-cl2 をマスターに切り替える必要があります。切り替えはコマンドで実施します。


■ 構文

```
# /etc/init.d/corosync stop
# /etc/init.d/rsyslog start
# /etc/init.d/corosync start
```

このコマンドはクラスタの制御をしている corosync の再起動を実施します。
これによりクラスタの切り替えが可能です。corosync を起動させる前に、rsyslog も起動させます。また、発行するのはマスターになりますので xmon-cl1 で実施します。

3.1 切り替えコマンドの発行

実際に切り替えコマンドの発行してみます。

 **発行するホストを間違えないように注意してください**

```
[root@xmon-cl1 ~]# /etc/init.d/corosync stop
Signaling Corosync Cluster Engine (corosync) to terminate: [ OK ]
Waiting for corosync services to unload:..... [ OK ]
[root@xmon-cl1 ~]# /etc/init.d/rsyslog start
Starting system logger:
[root@xmon-cl1 ~]# /etc/init.d/corosync start
Starting Corosync Cluster Engine (corosync): [ OK ]
```

停止中

起動中

再起動が完了すると上記の三行が [OK] となります。
rsyslog は「Starting system logger:」となります。
確認方法は第 2 章で説明しているコマンドで実施します。

3.2 切り替え後の確認 (xmon-cl1 スレーブ)

クラスタの状態を確認します。

```
[root@xmon-cl1 ~]# crm_mon -n1
=====
~ 中略 ~
Current DC: xmon-cl2 - partition with quorum
~ 中略 ~
=====
Node xmon-cl1: online
```

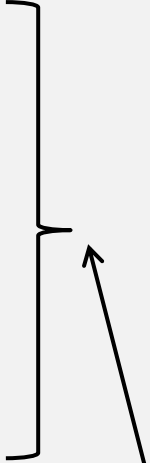
マスターが xmon-cl2 になっている事が表示。

スレーブの xmon-cl1 では DRBD のみ動作している。

```

res_drbd:0      (ocf::linbit:drbd) Started
Node xmon-cl2: online
res_snmpptt     (lsb:snmpptt) Started
res_ln-x-mon    (lsb:ln-x-mon) Started
res_nsca        (lsb:nsca) Started
res_vip (ocf::heartbeat:IPAddr2) Started
res_filesystem  (ocf::heartbeat:Filesystem) Started
res_httpd       (lsb:httpd) Started
res_postgresql  (lsb:postgresql) Started
res_drbd:1      (ocf::linbit:drbd) Master
res_x-mon       (lsb:x-mon) Started
res_snmptrapd   (lsb:snmptrapd) Started

```



スレーブの xmon-cl1 では DRBD のみ動作している。

データ領域の確認をします。

```
[root@xmon-cl1 ~]# service drbd status
```

～中略～

```

m:res  cs          ro          ds          p mounted fstype
0:r0   Connected   Secondary/Primary UpToDate/UpToDate C

```

自身がセカンダリになり、マウントしていない事が確認出来ます。

フローティング IP アドレスが割り当てられていない事も確認してみましょう。

```
[root@xmon-cl1 ~]# /sbin/ifconfig eth0:0
```

```

eth0:0   Link encap:Ethernet  HWaddr 00:0C:29:AB:3A:25
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1

```

システムから見てもマウントされていない事を確認します。

```
[root@xmon-cl1 ~]# df -Th
```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
/dev/sda5	ext4	9.7G	901M	8.3G	10%	/
tmpfs	tmpfs	3.9G	17M	3.9G	1%	/dev/shm
/dev/sda1	ext4	485M	49M	412M	11%	/boot
/dev/sda6	ext4	9.7G	1.1G	8.2G	12%	/usr
/dev/sda2	ext4	20G	348M	18G	2%	/var

3.3 切り替え後の確認 (xmon-cl2 マスター)

切り替わり、マスターとなった xmon-cl2 を確認してみます。

```
[root@xmon-cl2 ~]# crm_mon -n1
```

```
=====
```

～中略～

```
Current DC: xmon-cl2 - partition with quorum
```

～中略～

```
=====
```

```
Node xmon-cl1: online
```

```
res_drbd:0 (ocf::linbit:drbd) Started
```

```
Node xmon-cl2: online
```

```
res_snmpptt (lsb:snmpptt) Started
```

```
res_ln-x-mon (lsb:ln-x-mon) Started
```

```
res_nsca (lsb:nsca) Started
```

```
res_vip (ocf::heartbeat:IPAddr2) Started
```

```
res_filesystem (ocf::heartbeat:Filesystem) Started
```

```
res_httpd (lsb:httpd) Started
```

```
res_postgresql (lsb:postgresql) Started
```

```
res_drbd:1 (ocf::linbit:drbd) Master
```

```
res_x-mon (lsb:x-mon) Started
```

```
res_snmptrapd (lsb:snmptrapd) Started
```

xmon-cl1 と同じ内容が表示され、xmon-cl2 で各リソースが動作している事が確認できます。

データ領域を確認します。

```
[root@xmon-cl2 ~]# service drbd status
```

～中略～

```
m:res cs ro ds p mounted fstype
0:r0 Connected Primary/Secondary UpToDate/UpToDate C /data ext4
```

xmon-cl2 がプライマリとなり、/data をマウントしている事が表示されました。

フローティング IP アドレスを確認します。

```
[root@xmon-cl2 ~]# /sbin/ifconfig eth0:0
```

```
eth0:0 Link encap:Ethernet HWaddr 00:0C:29:0F:B3:D9
```

```
inet addr:192.168.19.133 Bcast:192.168.19.255 Mask:255.255.255.0
```

```
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
```

eth0:0 にフローティング IP アドレスが割り当てられました。

システムからマウントを確認します。

```
[root@xmon-cl2 ~]# df -Th
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/sda5       ext4      9.7G  529M   8.7G   6% /
tmpfs           tmpfs     3.9G   17M   3.9G   1% /dev/shm
/dev/sda1       ext4      485M   49M   412M  11% /boot
/dev/sda6       ext4      9.7G  1.1G   8.2G  12% /usr
/dev/sda2       ext4      20G   333M   18G   2% /var
/dev/drbd0      ext4     398G   1.1G  377G   1% /data
```

/data が正常にマウントされた事が確認出来ました。

3.4 X-MON にて確認

X-MON の監視で確認してみましょう。

図 X-MON



「Master node:xmon-cl2」となっています。

これで正常に切り替えが出来ている事が確認出来ました。

4 切り替えたクラスタのマスターをデフォルトに戻す

切り替えたクラスタで稼働後、xmon-cl2 がマスターとなり新しいデータが蓄積される状態となります。停止していた xmon-cl1 を起動後も自動ではフェイルバックせず xmon-cl2 をマスターのまま稼働させる事が可能です。

しかし、その状態で xmon-cl1（スレーブ）と xmon-cl2（マスター）が同時に再起動してしまった場合、仕様により xmon-cl1 が優先してマスターとなります。

そうすると、xmon-cl2（スレーブ）の方が新しいデータを蓄積しているため xmon-cl1（マスター）からデータを同期しようとしても DRBD の制御によりデータは上書きしない動作が発生します。この状態がデータ同期機能の切断状態です。

この状態を防ぐため、切り替え後に xmon-cl1 が稼働可能状態になった場合は xmon-cl1 をマスターに手動で戻す作業が必要となります。

切り替えはコマンドで実施します。

■ 構文

```
# /etc/init.d/corosync stop
# /etc/init.d/rsyslog start
# /etc/init.d/corosync start
```

このコマンドはクラスタの制御をしている corosync の再起動を実施します。

これによりクラスタの切り替えが可能です。corosync を起動させる前に、rsyslog も起動させます。また、発行するのはマスターになりますので xmon-cl2 で実施します。

4.1 切り替えコマンドの発行



作業の事前に xmon-cl1 がスレーブ、xmon-cl2 がマスターであることを確認してから作業を実施してください。

図 X-MON で確認

X-MON (X-MON)	Maser Node	正常(OK)	2012-11-21 16:35:26	9日と09時間29分59秒	1/3	Master node: xmon-cl2
	PING	正常(OK)	2012-11-21 16:33:59	9日と09時間28分30秒	1/3	PING OK - Packet loss = 0%, RTA = 0.03 ms

Master node: xmon-cl2

コマンドを発行します。

```
[root@xmon-cl2 ~]# /etc/init.d/corosync stop
Signaling Corosync Cluster Engine (corosync) to terminate: [ OK ]
Waiting for corosync services to unload:..... [ OK ]
[root@xmon-cl2 ~]# /etc/init.d/rsyslog start
Starting system logger:
[root@xmon-cl2 ~]# /etc/init.d/corosync start
Starting Corosync Cluster Engine (corosync): [ OK ]
```

停止中

起動中

再起動が完了すると上記の三行が[OK] となります。

rsyslog は「Starting system logger:」となります。

正常に切り替わったか確認していきます。

4.2 切り替え後の確認 (xmon-cl1 マスター)

クラスタの状態を確認します。

```
[root@xmon-cl1 ~]# crm_mon -n1
=====
~中略~
Current DC: xmon-cl1 - partition with quorum
~中略~
=====
Node xmon-cl1: online
    res_ln-x-mon    (lsb:ln-x-mon) Started
    res_snmpptt     (lsb:snmpptt) Started
    res_nsca        (lsb:nsca) Started
    res_vip (ocf::heartbeat:IPaddr2) Started
    res_filesystem  (ocf::heartbeat:Filesystem) Started
    res_httpd       (lsb:httpd) Started
    res_postgresql  (lsb:postgresql) Started
    res_drbd:0      (ocf::linbit:drbd) Master
    res_x-mon       (lsb:x-mon) Started
    res_snmptrapd   (lsb:snmptrapd) Started
Node xmon-cl2: online
    res_drbd:1      (ocf::linbit:drbd) Started
```

マスターが xmon-cl1 になっている事が表示。

マスターの xmon-cl1 では各リソースが動作している。

スレーブの xmon-cl2 では DRBD のみ動作している。

データ領域の確認

```
[root@xmon-cl1 ~]# service drbd status
```

～中略～

```
m:res  cs          ro          ds          p  mounted  fstype
0:r0   Connected  Primary/Secondary UpToDate/UpToDate C  /data    ext4
```

自身がプライマリになり、/data をマウントしている事が確認出来ます。

フローティング IP アドレスが eth0:0 に割り当てられている事を確認します。

```
[root@xmon-cl1 ~]# /sbin/ifconfig eth0:0
```

```
eth0:0  Link encap:Ethernet  HWaddr 00:0C:29:AB:3A:25
```

```
inet addr:192.168.19.133 Bcast:192.168.19.255 Mask:255.255.255.0
```

```
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

システムからマウントの確認もしてみましょう。

```
[root@xmon-cl1 ~]# df -Th
```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
/dev/sda5	ext4	9.7G	901M	8.3G	10%	/
tmpfs	tmpfs	3.9G	17M	3.9G	1%	/dev/shm
/dev/sda1	ext4	485M	49M	412M	11%	/boot
/dev/sda6	ext4	9.7G	1.1G	8.2G	12%	/usr
/dev/sda2	ext4	20G	348M	18G	2%	/var
/dev/drbd0	ext4	398G	1.1G	377G	1%	/data

/data が正常にマウントされている事が確認出来ました。

4.3 切り替え後の確認 (xmon-cl2 スレーブ)

切り替えコマンドを発行した xmon-cl2 がスレーブになったか確認します。

クラスタの状態の確認してみます。

```
[root@xmon-cl2 ~]# crm_mon -n1
```

```
=====
```

～中略～

```
Current DC: xmon-cl1 - partition with quorum
```

～中略～

```
=====
```

```
Node xmon-cl1: online
```

```
res_ln-x-mon (lsb:ln-x-mon) Started
```

```
res_snmpd      (lsb:snmpd) Started
res_nsca       (lsb:nsca) Started
res_vip (ocf::heartbeat:IPaddr2) Started
res_filesystem (ocf::heartbeat:Filesystem) Started
res_httpd      (lsb:httpd) Started
res_postgresql (lsb:postgresql) Started
res_drbd:0     (ocf::linbit:drbd) Master
res_x-mon      (lsb:x-mon) Started
res_snmptrapd  (lsb:snmptrapd) Started
Node xmon-cl2: online
res_drbd:1     (ocf::linbit:drbd) Started
```

xmon-cl1 と同じ内容が表示され、xmon-cl1 がマスターで各リソースが動作している事が確認出来ました。

データ領域の確認をしてみます。

```
[root@xmon-cl2 ~]# service drbd status
~中略~
m:res  cs          ro          ds          p mounted fstype
0:r0   Connected Secondary/Primary UpToDate/UpToDate C
```

自身がセカンダリになり、マウントしていない事が確認出来ます。

フローティング IP アドレスが割り当てられていない事も確認しましょう。

```
[root@xmon-cl2 ~]# /sbin/ifconfig eth0:0
eth0:0  Link encap:Ethernet HWaddr 00:0C:29:0F:B3:D9
        UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
```


システムからも/data をマウントしていない事を確認します。

```
[root@xmon-cl2 ~]# df -Th
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/sda5       ext4      9.7G  529M  8.7G   6% /
tmpfs           tmpfs     3.9G   17M  3.9G   1% /dev/shm
/dev/sda1       ext4      485M   49M  412M  11% /boot
/dev/sda6       ext4      9.7G  1.1G  8.2G  12% /usr
/dev/sda2       ext4      20G   333M  18G   2% /var
```

4.4 X-MON で確認

X-MON の監視で確認してみましょう。

図 X-MON

X-MON (X-MON)		Master Node	正常 (OK)	2012-11-21 16:54:00	9日と09時間48分 58秒	1/3	Master node: xmon-cl1
		PING	正常 (OK)	2012-11-21 16:53:59	9日と09時間47分 29秒	1/3	PING OK - Packet loss = 0%, RTA = 0.03 ms

Master node: xmon-cl1

「Master node:xmon-cl1」となっています。

これで正常に切り替えが出来ている事が確認出来ました。

切り替え後は xmon-cl1 がマスターで稼働しますので正常に監視が出来ている事を確認してください。

5 データ同期が外れた場合の復旧について

データの同期が外れるのはデータ領域を制御する DRBD に障害が発生したりすると発生します。

[2.2.3 障害が発生している場合](#) でも説明させて頂いた内容となります。

この復旧について説明していきます。

また、データ同期が外れる場合は xmon-cl1 がマスター、xmon-cl2 がスレーブの場合とします。単体では発生せず、マスターを xmon-cl2 に切り替えを実施していても xmon-cl1 が稼働可能になった場合は xmon-cl1 をマスターに切り替えを実施するためです。

5.1 マスター、スレーブともに StandAlone 状態

障害が発生した時は xmon-cl1、xmon-cl2 とともに孤立した状態となり、自分のデータを守ろうとする動作をします。その際の状態が StandAlone となります。

xmon-cl1（マスター）の状態を確認します。

```
[root@xmon-cl1 ~]# service drbd status
```

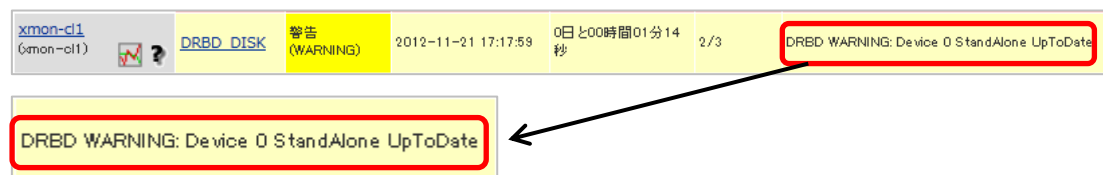
～中略～

```
m:res cs ro ds p mounted fstype
0:r0 StandAlone Primary/Unknown UpToDate/DUnknown r----- ext4
```

StandAlone 状態となり、相手（xmon-cl2）と DRBD 接続が出来ないため、Primary/Unknown 状態となり、データの状態も UpToDate/Dunknown 状態となっています。

この状態では X-MON では WARNING を検知します。

図 X-MON



xmon-cl2（スレーブ）の状態を確認します。

```
[root@xmon-cl2 ~]# service drbd status
```

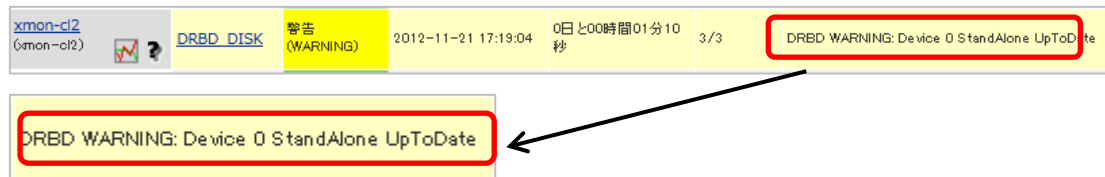
～中略～

```
m:res cs ro ds p mounted fstype
0:r0 StandAlone Secondary/Unknown UpToDate/DUnknown r-----
```

マスターと同じく StandAlone 状態で相手（xmon-cl1）と DRBD 接続が出来ないため、Secondary/Unknown 状態となり、データの状態も UpToDate/Dunknown 状態となっています。

この状態では X-MON では WARNING を検知します

図 X-MON




復旧の流れとしては、両ホストとも「StandAlone」のままではデータの同期が出来ないため、スレーブの xmon-cl2 には「データを守る必要がない」ということを設定します。設定はコマンドで実施します。コマンド発行後、xmon-cl2 は **WFConnection** の接続待ち状態となります。その状態で xmon-cl1 にて「自分のデータを使用し、相手と接続する」という設定をコマンドで実施します。

まずは、xmon-cl2 に「データを守る必要がない」という設定するコマンドを発行します。

■ 構文

```
# drbdadm -- --discard-my-data connect r0
```

それでは xmon-cl2（スレーブ）にて発行してみます。

 **発行するホストを間違えないように注意してください**

```
[root@xmon-cl2 ~]# drbdadm -- --discard-my-data connect r0
[root@xmon-cl2 ~]#
```

発行し、エラーメッセージなどが表示されずにプロンプトへ返ると発行完了です。

データ領域の確認をします。

```
[root@xmon-cl2 ~]# service drbd status
~中略~
m:res cs ro ds p mounted fstype
0:r0 WFConnection Secondary/Unknown UpToDate/DUnknown C
```

正常にコマンドが発行されると WFConnection となり、接続待ち状態となります。

この状態では X-MON では CRITICAL を検知しますが正常な動作です。

図 X-MON



xmon-cl2 の DRBD_DISK 監視でステータス情報が

「DRBD CRITICAL:Device 0 Wfconnection UpToDate」と表示されます。

xmon-cl1（マスター）では StandAlone 状態のままになりますが正常です。

```
[root@xmon-cl1 ~]# service drbd status
```

～中略～

m:res	cs	ro	ds	p	mounted	fstype
0:r0	StandAlone	Primary/Unknown	UpToDate/DUnknown	r-----	ext4	

続いて、マスターで復旧作業を実施しますので次章へ進みます。

5.2 マスターが StandAlone,スレーブが Wfconnection 状態

続いて、xmon-cl1（マスター）に「自分のデータを使用し、相手と接続する」という設定のコマンドを発行します。

■構文

```
# drbdadm connect r0
```

それでは xmon-cl1（マスター）で発行します。



発行するホストを間違えないように注意してください

```
[root@xmon-cl1 ~]# drbdadm connect r0
```

```
[root@xmon-cl1 ~]#
```

発行し、エラーメッセージなどが表示されずにプロンプトへ返ると発行完了です。

データ領域の確認をします。

```
[root@xmon-cl1 ~]# service drbd status
```

～中略～

m:res	cs	ro	ds	p	mounted	fstype
0:r0	Connected	Primary/Secondary	UpToDate/UpToDate	C	/data	ext4

DRBD 接続が正常な状態になったので、正常なプライマリ時の表示になります。

続いて xmon-cl2（スレーブ）でも確認してみましょう。

```
[root@xmon-cl2 ~]# service drbd status
```

～中略～

m:res	cs	ro	ds	p	mounted	fstype
0:r0	Connected	Secondary/Primary	UpToDate/UpToDate	C		

DRBD 接続が正常な状態になったので、正常なセカンダリ時の表示になります。

X-MON の監視でも復旧を確認します。

xmon-cl1 (xmon-cl1)		DRBD_DISK	正常(OK)	2012-11-21 17:24:59	0日と00時間00分38秒	1/3	DRBD OK: Device 0 Connected UpToDate
xmon-cl2 (xmon-cl2)		DRBD_DISK	正常(OK)	2012-11-21 17:25:31	0日と00時間01分10秒	1/3	DRBD OK: Device 0 Connected UpToDate

DRBD OK: Device 0 Connected UpToDate

xmon-cl1、xmon-cl2 とともに DRBD_DISK 監視が正常となり、
「DRBD OK:Device 0 Connected UpToDate」のステータス情報となります。
これで復旧作業は完了です。

確認後は念のため、他に影響が出ていないかクラスタの状態、フローティング IP アドレスの割り当て、/data のマウント状態を確認してください。

6 機器の障害について

クラスタのハードウェアについて、X-MON にて障害監視を実施しています。
監視の種類は各ホスト共通です。

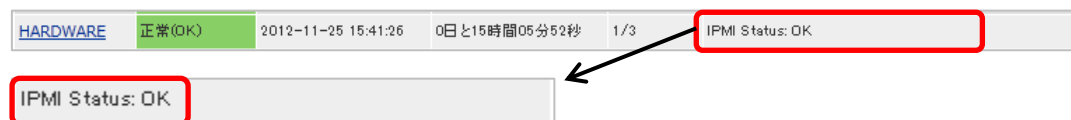
6.1 HARDWARE 監視

IPMI を通じてハードウェアの監視を行っています。

ファンや電源に異常が発生した場合、この HARDWARE 監視が障害を検知します。

正常な場合は図のように「IPMI Status: OK」が表示されます。

図 正常な場合



また、サービス情報画面のサービス詳細にてパフォーマンスデータも確認出来ます。

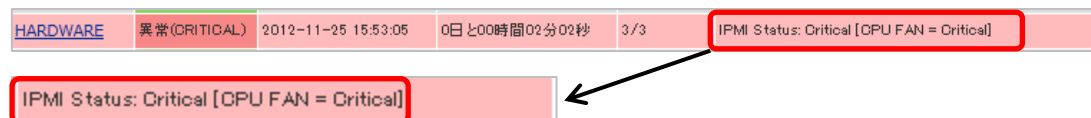
図 パフォーマンスデータ



異常な場合は CRITICAL と共にステータス情報に内容が表示されます。

例では、FAN に異常が発生した際のステータス情報です。

図 障害を検知した場合



サービス情報画面のサービス詳細にて確認してみます。

図 パフォーマンスデータ

障害対応ガイド		サービス詳細	ドキュメント	構成情報	イベントログ	通知履歴
現在の状態:	異常(CRITICAL) (0日間と 00時間00分08秒前より継続しています。)					
ステータス情報:	IPMI Status: Critical [CPU FAN = Critical]					
パフォーマンスデータ:	'+5V'=5.080000 '+5VSB'=5.080000 '+12V'=12.000000 '+V CORE'=0.640000 '+3.3V'=3.360000 'CPU FAN'=0.000000 'SYS FAN'=9310.000000 'AUX FAN'=9590.000000 'CPU Temp'=22.000000 'System TEMP'=34.000000 'AUX TEMP'=34.000000					

FAN の異常が確認出来ます。

故障が発生した場合は弊社までご連絡をお願いします。

6.2 RAID 監視

ハードウェアに搭載されているハードディスクと RAID 状態を監視しています。

異常が発生した場合、障害を検知します。

正常な場合は図のように「Logical Device 1 Optimal,Controller OK」と表示されます。

図 正常な場合

RAID	正常(OK)	2012-11-25 15:33:12	0日と15時間20分14秒	1/3	Logical Device 1 Optimal,Controller OK
Logical Device 1 Optimal,Controller OK					

異常な場合は CRITICAL と共にステータス情報に内容が表示されます。

例では、ハードディスクに異常が発生した際のステータス情報です。

図 障害を検知した場合

RAID	異常(CRITICAL)	2012-11-25 16:02:51	0日と00時間01分38秒	2/3	Logical Device 1 Degraded,Controller OK
Logical Device 1 Degraded,Controller OK					

RAID コントローラは正常なので「Controller OK」が表示されますが、ハードディスクが故障した事を示す「Logical Device 1 Degraded」が表示されます。

RAID 監視ではパフォーマンス情報欄には表示がされないため、ステータス情報欄で内容をご確認ください。

故障が発生した場合は弊社までご連絡をお願いします。

7 OS の設定変更方法

OS の設定変更方法について解説します。

7.1 IP アドレス変更手順

IP アドレスが変更になった場合、物理アドレス以外にも設定変更が必要となります。

変更する IP アドレスによって設定変更する箇所も異なります。

変更作業においてはリモートから実施すると IP アドレス変更後接続できなくなりますので、サーバ筐体にモニタとキーボードを接続して作業を実施してください。

また、作業中は X-MON が停止致しますので監視がストップします。

7.1.1 X-MON のマスターの確認

通常、X-MON のマスターは xmon-cl1 で動作しています。

本手順に関してもマスターは xmon-cl1 で動作している事を前提として記載しています。マスターに関しては X-MON の監視画面で確認出来ます。クラスタが切り替えを行っている環境（xmon-cl2 がマスター）の場合は、次章以降のマスターの部分を読み替えて作業を実施してください。

X-MON (X-MON)		Master Node	正常(OK)	2012-11-21 16:54:00	9日と09時間48分58秒	1/3	Master node: xmon-cl1
		PING	正常(OK)	2012-11-21 16:53:59	9日と09時間47分29秒	1/3	PING OK - Packet loss = 0%, RTA = 0.03 ms

7.1.2 サービス用 IP アドレスの変更手順

クラスタ X-MON のサービス用 IP アドレスを変更する場合は下記の手順となります。

- ・ クラスタ同期制御の corosync サービスを停止
- ・ 物理 IP アドレスの変更
- ・ クラスタ同期用設定変更
- ・ フローティング IP アドレスの変更

サービス用 IP アドレスは例では eth0 に設定されている前提で解説を行います。

他の NIC に割り当てられている場合は読み替えて確認ください。

本手順では下記を変更として記載しております。

- ・ xmon-cl1 の eth0 に割り当てられている IP アドレス 192.168.24.131/24 を 172.16.32.131/24 へ変更
- ・ xmon-cl2 の eth0 に割り当てられている IP アドレス 192.168.24.132/24 を 172.16.32.132/24 へ変更
- ・ GATEWAY に割り当てられている IP アドレス 192.168.24.1 を 172.16.32.1 へ変更
- ・ フローティング IP アドレスに割り当てられている IP アドレス 192.168.24.133/24 を 172.16.32.133/24 へ変更

7.1.2.1 corosync を停止

設定変更前にクラスタ同期制御を行う corosync サービスを停止させます。

デフォルトでは xmon-cl1 がマスター、xmon-cl2 がスレーブで動作しています。

xmon-cl2 から停止させます。



発行するホストを間違えないように注意してください

corosync サービスを二台とも停止した時点で X-MON が停止しますので監視が停止し、管理画面にもアクセス出来なくなります。

```
[root@xmon-cl2 ~]# /etc/init.d/corosync stop
```

```
Signaling Corosync Cluster Engine (corosync) to terminate:
```

```
[ OK ]
```

停止中

```
Waiting for corosync services to unload:.....
```

```
[ OK ]
```

次に xmon-cl1 で corosync を停止させます。

```
[root@xmon-cl1 ~]# /etc/init.d/corosync stop
```

```
Signaling Corosync Cluster Engine (corosync) to terminate:
```

```
[ OK ]
```

停止中

```
Waiting for corosync services to unload:.....
```

```
[ OK ]
```

クラスタ制御が停止しますので、確認コマンドを発行しても確認項目は表示されなくなります。

```
[root@ xmon-cl1 ~]# crm_mon -n1
```

```
Connection to cluster failed: connection failed
```

```
[root@ xmon-cl1 ~]# service drbd status
```

```
drbd driver loaded OK; device status:
```

```
version: 8.4.2 (api:1/proto:86-101)
```

```
GIT-hash: 7ad5f850d711223713d6dcadc3dd48860321070c build by dag@Build64R6,  
2012-09-06 08:16:10
```

```
m:res cs ro ds p mounted fstype
```

また、この時点でフローティング IP も制御から外れるため割り当てがなくなります。

```
[root@xmon-cl1 ~]# /sbin/ifconfig eth0:0
```

```
eth0:0    Link encap:Ethernet  HWaddr 00:0C:29:0F:B3:D9
```

```
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

7.1.2.2 物理 IP アドレスの変更

corosync サービスの停止を確認後、物理 IP アドレスを変更します。

■対象ファイル

```
/etc/sysconfig/network-scripts/ifcfg-eth0
```

■対象機器

xmon-cl1,xmon-cl2 ともに変更を実施します

バックアップ用ディレクトリを作成し、バックアップを取得します。

```
[root@xmon-cl1 ~]# mkdir ~/backup
[root@xmon-cl1 ~]# cp -vip /etc/sysconfig/network-scripts/ifcfg-eth0 ~/backup/
`/etc/sysconfig/network-scripts/ifcfg-eth0' -> `/root/backup/ifcfg-eth0'
```

xmon-cl1、xmon-cl2 ともに設定ファイルの書き換えを実施してください。

設定を変更します。

```
[root@xmon-cl1 ~]# vi /etc/sysconfig/network-scripts/ifcfg-eth0
```

■編集前

```
IPADDR=192.168.24.131
NETMASK=255.255.255.0
GATEWAY=192.168.24.1
```

■編集後

```
IPADDR=172.16.32.131
NETMASK=255.255.255.0
GATEWAY=172.16.32.1
```

変更が完了したら、変更を反映させます。

```
[root@xmon-cl1 ~]# ifdown eth0
[root@xmon-cl1 ~]# ifup eth0
```

反映を確認します。

```
[root@xmon-cl1 ~]# /sbin/ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0C:29:0F:B3:D9
          inet addr:172.16.32.131 Bcast:172.16.32.255 Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
[root@xmon-cl1 ~]# /sbin/route -n
0.0.0.0    172.16.32.1  0.0.0.0    UG    0    0    0 eth0
```

同様に、xmon-cl2 でも変更を実施してください。

7.1.2.3 corosync の設定変更

クラスタ同期制御を行う corosync の設定変更を行います。

■対象ファイル

/etc/corosync/corosync.conf

■対象機器

xmon-cl1,xmon-cl2 ともに変更を実施します。変更箇所は共通です。

バックアップを取得します。

```
[root@xmon-cl1 ~]# cp -vip /etc/corosync/corosync.conf ~/backup/  
`/etc/corosync/corosync.conf' -> `/root/backup/corosync.conf'
```

xmon-cl1、xmon-cl2 ともに設定ファイルの書き換えを実施してください。

設定を変更します。ringnumber: 0 の部分の bindnetaddr の部分となります。

IP アドレスではなく **ネットワークアドレス**が記載されていますので

変更後のネットワークアドレスを記載してください。

```
[root@xmon-cl1 ~]# vi /etc/corosync/corosync.conf
```

■編集前

```
interface {  
    ringnumber: 0  
    bindnetaddr: 192.168.24.0  
    mcastaddr: 226.94.1.1  
    mcastport: 5405  
    ttl: 1  
}
```

■編集後

```
interface {  
    ringnumber: 0  
    bindnetaddr: 172.16.32.0  
    mcastaddr: 226.94.1.1  
    mcastport: 5405  
    ttl: 1  
}
```

7.1.2.4 corosync を起動

設定変更が出来てから corosync を起動させます。起動させる前に、rsyslog を再起動させます。xmon-cl1 から起動させます。



発行するホストを間違えないように注意してください

```
[root@xmon-cl1 ~]# /etc/init.d/rsyslog restart
```

```
Starting system logger:
```

```
[root@xmon-cl1 ~]# /etc/init.d/corosync start
```

```
Starting Corosync Cluster Engine (corosync):
```

```
[ OK ]
```

起動中

次に xmon-cl2 でも corosync を起動させます。

```
[root@xmon-cl2 ~]# /etc/init.d/rsyslog restart
```

```
Starting system logger:
```

```
[root@xmon-cl2 ~]# /etc/init.d/corosync start
```

起動中

```
Starting Corosync Cluster Engine (corosync):
```

[OK]

起動させて 20 秒ほどしたらデータの同期が完了しますので確認してください。

クラスタの状態を確認します。

```
[root@xmon-cl1 ~]# crm_mon -n1
```

```
=====
```

～中略～

```
Current DC: xmon-cl1 - partition with quorum
```

～中略～

```
=====
```

```
Node xmon-cl1: online
```

```
res_ln-x-mon    (lsb:ln-x-mon) Started
res_snmpptt     (lsb:snmpptt) Started
res_nsca        (lsb:nsca) Started
res_vip         (ocf::heartbeat:IPAddr2) Started
res_filesystem  (ocf::heartbeat:Filesystem) Started
res_httpd       (lsb:httpd) Started
res_postgresql  (lsb:postgresql) Started
res_drbd:0      (ocf::linbit:drbd) Master
res_x-mon       (lsb:x-mon) Started
res_snmptrapd   (lsb:snmptrapd) Started
```

```
Node xmon-cl2: online
```

```
res_drbd:1      (ocf::linbit:drbd) Started
```

マスターが xmon-cl1 になっている事が表示。

マスターの xmon-cl1 では各リソースが動作している。

スレーブの xmon-cl2 では DRBD のみ動作している。

データ領域の確認

```
[root@xmon-cl1 ~]# service drbd status
```

～中略～

```
m:res  cs      ro      ds      p  mounted  fstype
0:r0   Connected Primary/Secondary UpToDate/UpToDate C  /data    ext4
```

自身がプライマリになり、/data をマウントしている事が確認出来ます。

システムからマウントの確認もしてみましょう。

```
[root@xmon-cl1 ~]# df -Th
```

```
Filesystem    Type    Size  Used Avail Use% Mounted on
```

```
/dev/sda5    ext4    9.7G  901M  8.3G  10% /
tmpfs        tmpfs    3.9G   17M  3.9G   1% /dev/shm
/dev/sda1    ext4    485M   49M  412M  11% /boot
/dev/sda6    ext4    9.7G  1.1G  8.2G  12% /usr
/dev/sda2    ext4    20G   348M   18G   2% /var
/dev/drbd0   ext4    398G  1.1G  377G   1% /data
```

/data が正常にマウントされている事が確認出来ました。

フローティング IP アドレスはまだ変更されていません。

```
[root@xmon-cl1 ~]# /sbin/ifconfig eth0:0
eth0:0    Link encap:Ethernet  HWaddr 00:0C:29:AB:3A:25
          inet addr:192.168.24.133  Bcast:192.168.24.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

xmon-cl2 でも確認しておきます。

```
[root@xmon-cl2 ~]# crm_mon -n1
```

=====

～中略～

Current DC: xmon-cl1 - partition with quorum

～中略～

=====

Node xmon-cl1: online

```
res_ln-x-mon    (lsb:ln-x-mon) Started
res_snmpptt     (lsb:snmpptt) Started
res_nsca        (lsb:nsca) Started
res_vip (ocf::heartbeat:IPAddr2) Started
res_filesystem  (ocf::heartbeat:Filesystem) Started
res_httpd       (lsb:httpd) Started
res_postgresql  (lsb:postgresql) Started
res_drbd:0      (ocf::linbit:drbd) Master
res_x-mon       (lsb:x-mon) Started
res_snmptrapd    (lsb:snmptrapd) Started
```

Node xmon-cl2: online

```
res_drbd:1      (ocf::linbit:drbd) Started
```

xmon-cl1 と同じ内容が表示され、xmon-cl1 がマスターで各リソースが動作している事が確認出来ました。

データ領域の確認をしてみます。

```
[root@xmon-cl2 ~]# service drbd status
```

～中略～

```
m:res  cs          ro          ds          p  mounted  fstype
0:r0   Connected  Secondary/Primary UpToDate/UpToDate C
```

自身がセカンダリになり、マウントしていない事が確認出来ます。

フローティング IP アドレスが割り当てられていない事も確認しましょう。

```
[root@xmon-cl2 ~]# /sbin/ifconfig eth0:0
```

```
eth0:0    Link encap:Ethernet  HWaddr 00:0C:29:0F:B3:D9
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

システムからも/data をマウントしていない事を確認します。

```
[root@xmon-cl2 ~]# df -Th
```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
/dev/sda5	ext4	9.7G	529M	8.7G	6%	/
tmpfs	tmpfs	3.9G	17M	3.9G	1%	/dev/shm
/dev/sda1	ext4	485M	49M	412M	11%	/boot
/dev/sda6	ext4	9.7G	1.1G	8.2G	12%	/usr
/dev/sda2	ext4	20G	333M	18G	2%	/var

7.1.2.5 フローティング IP アドレスの変更

最後にフローティング IP アドレスの変更を行います。

変更はマスター (xmon-cl1) でのみ行います。

■構文

```
# crm configure edit
```

■対象機器

xmon-cl1 のみ。変更すると xmon-cl2 にも反映されます。

コマンドを発行すると、vi の編集画面となりますのでフローティング IP アドレスを修正します。

```
[root@xmon-cl1 ~]# crm configure edit
```

■編集前

```
primitive res_vip ocf:heartbeat:IPaddr2 ¥
```

```
    params ip="192.168.24.133" cidr_netmask="24" nic="eth0:0" ¥
```

```
op monitor interval="20s"
```

■編集後

```
primitive res_vip ocf:heartbeat:IPAddr2 ¥  
    params ip="172.16.32.133" cidr_netmask="24" nic="eth0:0" ¥  
    op monitor interval="20s"
```

サブネットマスクが変更している場合は `cidr_netmask="24"` の部分を変更してください。

変更が完了すれば「:wq」で編集画面を抜けてプロンプトに戻ります。

保存するだけで反映が実施され、20 秒程度で新しい IP アドレスが付与されます。

```
[root@xmon-cl1 ~]# /sbin/ifconfig eth0:0  
eth0:0    Link encap:Ethernet  HWaddr 00:0C:29:AB:3A:25  
          inet addr:172.16.32.133  Bcast:172.16.32.255  Mask:255.255.255.0  
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

新しいフローティング IP アドレスで X-MON にアクセスできるか確認してください。
以上でサービス用 IP アドレスの変更手順は終了です。

7.1.2.6 同じネットワーク内で IP アドレスが変更する場合

同じネットワーク内での変更の場合は物理 IP アドレスの変更後、corosync の設定変更は必要ありません。そのため、手順としては、以下の 3 つの手順を実施してください。

- | | |
|---------|---------------|
| 7.1.2.1 | corosync を停止 |
| 7.1.2.2 | 物理 IP アドレスの変更 |
| 7.1.2.4 | corosync を起動 |

フローティング IP アドレスの変更が必要な場合は、

7.1.2.5	フローティング IP アドレスの変更
---------	--------------------

にて変更を実施してください。

7.1.3 同期用 IP アドレスの変更手順

クラスタ X-MON の同期用 IP アドレスを変更する場合は下記の手順となります。

- ・ クラスタ同期制御の corosync サービスを停止
- ・ 物理 IP アドレスの変更
- ・ /etc/hosts の変更

- ・ DRBD の設定変更
- ・ クラスタ同期用設定変更
- ・ /etc/opt/nrpe/nrpe.cfg の設定変更
- ・ 監視対象 IP アドレスの変更

同期用 IP アドレスは例では eth1 に設定されている前提で解説を行います。

他の NIC に割り当てられている場合は読み替えて確認ください。

本手順では下記を変更として記載しております。

- ・ xmon-cl1 の eth1 に割り当てられている IP アドレス 10.0.0.1/24 を 172.16.16.1/24 へ変更
- ・ xmon-cl2 の eth1 に割り当てられている IP アドレス 10.0.0.2/24 を 172.16.16.2/24 へ変更

7.1.3.1 corosync を停止

設定変更前にクラスタ同期制御を行う corosync サービスを停止させます。

デフォルトでは xmon-cl1 がマスター、xmon-cl2 がスレーブで動作しています。

xmon-cl2 から停止させます。



発行するホストを間違えないように注意してください

corosync サービスを二台とも停止した時点で X-MON が停止しますので監視が停止し、管理画面にもアクセス出来なくなります。

```
[root@xmon-cl2 ~]# /etc/init.d/corosync stop
Signaling Corosync Cluster Engine (corosync) to terminate: [ OK ]
Waiting for corosync services to unload:..... [ OK ]
```

停止中

次に xmon-cl1 で corosync を停止させます。

```
[root@xmon-cl1 ~]# /etc/init.d/corosync stop
Signaling Corosync Cluster Engine (corosync) to terminate: [ OK ]
Waiting for corosync services to unload:..... [ OK ]
```

停止中

クラスタ制御が停止しますので、確認コマンドを発行しても確認項目は表示されなくなります。

```
[root@ xmon-cl1 ~]# crm_mon -n1
Connection to cluster failed: connection failed
[root@ xmon-cl1 ~]# service drbd status
drbd driver loaded OK; device status:
version: 8.4.2 (api:1/proto:86-101)
GIT-hash: 7ad5f850d711223713d6dcadc3dd48860321070c build by dag@Build64R6,
2012-09-06 08:16:10
```

```
m:res cs ro ds p mounted fstype
```

また、この時点でフローティング IP も制御から外れるため割り当てがなくなります。

```
[root@xmon-cl1 ~]# /sbin/ifconfig eth0:0
eth0:0    Link encap:Ethernet  HWaddr 00:0C:29:0F:B3:D9
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

7.1.3.2 物理 IP アドレスの変更

corosync サービスの停止を確認後、物理 IP アドレスを変更します。

■対象ファイル

/etc/sysconfig/network-scripts/ifcfg-eth1

■対象機器

xmon-cl1,xmon-cl2 ともに変更を実施します

バックアップ用ディレクトリを作成し、バックアップを取得します。

```
[root@xmon-cl1 ~]# mkdir ~/backup
[root@xmon-cl1 ~]# cp -vip /etc/sysconfig/network-scripts/ifcfg-eth1 ~/backup/
`/etc/sysconfig/network-scripts/ifcfg-eth1' -> `/root/backup/ifcfg-eth1'
```

xmon-cl1、xmon-cl2 ともに設定ファイルの書き換えを実施してください。

設定を変更します。

```
[root@xmon-cl1 ~]# vi /etc/sysconfig/network-scripts/ifcfg-eth1
```

■編集前

```
IPADDR=10.0.0.1
NETMASK=255.255.255.0
```

■編集後

```
IPADDR=172.16.16.1
NETMASK=255.255.255.0
```

変更が完了したら、変更を反映させます。

```
[root@xmon-cl1 ~]# ifdown eth1
[root@xmon-cl1 ~]# ifup eth1
```

反映を確認します。

```
[root@xmon-cl1 ~]# /sbin/ifconfig eth1
eth1      Link encap:Ethernet  HWaddr 00:0C:29:0F:B3:D9
          inet addr:172.16.16.1  Bcast:172.16.16.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

同様に、xmon-cl2 でも変更を実施してください。

7.1.3.3 /etc/hosts ファイルの変更

/etc/hosts ファイルに名前解決のための設定が記載されていますので変更します。

■対象ファイル

/etc/hosts

■対象機器

xmon-cl1,xmon-cl2 ともに変更を実施します。変更箇所は共通です。

バックアップを取得します。

```
[root@xmon-cl1 ~]# cp -vip /etc/hosts ~/bakup/  
`/etc/hosts' -> `/root/backup/hosts'
```

xmon-cl1、xmon-cl2 ともに設定ファイルの書き換えを実施してください。

設定を変更します。

```
[root@xmon-cl1 ~]# vi /etc/hosts
```

■編集前

10.0.0.1	xmon-cl1
10.0.0.2	xmon-cl2

■編集後

172.16.16.1	xmon-cl1
172.16.16.2	xmon-cl2

編集のみでサービスの再起動などはありません。

7.1.3.4 DRBD の設定変更

データの制御する DRDB の設定変更を行います。

■対象ファイル

/etc/drbd.d/data.res

■対象機器

xmon-cl1,xmon-cl2 ともに変更を実施します。変更箇所は共通です。

バックアップを取得します。

```
[root@xmon-cl1 ~]# cp -vip /etc/drbd.d/data.res ~/bakup/  
`/etc/drbd.d/data.res' -> `/root/bakup/data.res'
```

xmon-cl1、xmon-cl2 ともに設定ファイルの書き換えを実施してください。

設定を変更します。

```
[root@xmon-cl1 ~]# vi /etc/drbd.d/data.res
```

変更した IP アドレスに変更させます。

変更箇所は address の部分が二か所となります。

xmon-cl1,xmon-cl2 ともに変更内容は同じです。

■編集前

```
on xmon-cl1 {  
    device /dev/drbd0;  
    disk /dev/sda3;  
    address 10.0.0.1:7788;  
    meta-disk internal;  
}  
on xmon-cl2 {  
    device /dev/drbd0;  
    disk /dev/sda3;  
    address 10.0.0.2:7788;  
    meta-disk internal;  
}
```

■編集後

```
on xmon-cl1 {  
    device /dev/drbd0;  
    disk /dev/sda3;  
    address 172.16.16.1:7788;  
    meta-disk internal;  
}  
on xmon-cl2 {  
    device /dev/drbd0;  
    disk /dev/sda3;  
    address 172.16.16.2:7788;  
    meta-disk internal;  
}
```

編集のみでサービスの再起動などはありません。

7.1.3.5 corosync の設定変更

クラスタ同期制御を行う corosync の設定変更を行います。

■対象ファイル

/etc/corosync/corosync.conf

■対象機器

xmon-cl1, xmon-cl2 ともに変更を実施します。変更箇所は共通です。

バックアップを取得します。

```
[root@xmon-cl1 ~]# cp -vip /etc/corosync/corosync.conf ~/backup/  
`/etc/corosync/corosync.conf' -> `/root/backup/corosync.conf'
```

xmon-cl1、xmon-cl2 ともに設定ファイルの書き換えを実施してください。

設定を変更します。ringnumber: 1 の部分の bindnetaddr の部分となります。

IP アドレスではなくネットワークアドレスが記載されていますので

変更後の**ネットワークアドレス**を記載してください。

```
[root@xmon-cl1 ~]# vi /etc/corosync/corosync.conf
```

■編集前

```
interface {  
    ringnumber: 0  
    bindnetaddr: 10.0.0.0  
    mcastaddr: 226.94.1.1  
    mcastport: 5405  
    ttl: 1  
}
```

■編集後

```
interface {  
    ringnumber: 0  
    bindnetaddr: 172.16.16.0  
    mcastaddr: 226.94.1.1  
    mcastport: 5405  
    ttl: 1  
}
```

7.1.3.6 corosync を起動

設定変更が出来てから corosync を起動させます。**起動させる前に、rsyslog を再起動させます。** xmon-cl1 から起動させます。



発行するホストを間違えないように注意してください

```
[root@xmon-cl1 ~]# /etc/init.d/rsyslog restart
Starting system logger:
[root@xmon-cl1 ~]# /etc/init.d/corosync start
Starting Corosync Cluster Engine (corosync):
```

起動中

OK

次に xmon-cl2 でも corosync を起動させます。

```
[root@xmon-cl2 ~]# /etc/init.d/rsyslog restart
Starting system logger:
[root@xmon-cl2 ~]# /etc/init.d/corosync start
Starting Corosync Cluster Engine (corosync):
```

起動中

OK

起動させて 20 秒ほどしたらデータの同期が完了しますので確認してください。

クラスタの状態を確認します。

```
[root@xmon-cl1 ~]# crm_mon -n1
=====
~中略~
Current DC: xmon-cl1 - partition with quorum
~中略~
=====
Node xmon-cl1: online
    res_ln-x-mon    (lsb:ln-x-mon) Started
    res_snmpptt     (lsb:snmpptt) Started
    res_nsca        (lsb:nsca) Started
    res_vip         (ocf::heartbeat:IPAddr2) Started
    res_filesystem  (ocf::heartbeat:Filesystem) Started
    res_httpd       (lsb:httpd) Started
    res_postgresql  (lsb:postgresql) Started
    res_drbd:0      (ocf::linbit:drbd) Master
    res_x-mon       (lsb:x-mon) Started
    res_snmptrapd   (lsb:snmptrapd) Started
Node xmon-cl2: online
    res_drbd:1      (ocf::linbit:drbd) Started
```

マスターが xmon-cl1 になっている事が表示。

マスターの xmon-cl1 では各リソースが動作している。

スレーブの xmon-cl2 では DRBD のみ動作している。

データ領域の確認

```
[root@xmon-cl1 ~]# service drbd status
```

～中略～

m:res	cs	ro	ds	p	mounted	fstype
0:r0	Connected	Primary/Secondary	UpToDate/UpToDate	C	/data	ext4

自身がプライマリになり、/data をマウントしている事が確認出来ます。

システムからマウントの確認もしてみましょう。

```
[root@xmon-cl1 ~]# df -Th
```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
/dev/sda5	ext4	9.7G	901M	8.3G	10%	/
tmpfs	tmpfs	3.9G	17M	3.9G	1%	/dev/shm
/dev/sda1	ext4	485M	49M	412M	11%	/boot
/dev/sda6	ext4	9.7G	1.1G	8.2G	12%	/usr
/dev/sda2	ext4	20G	348M	18G	2%	/var
/dev/drbd0	ext4	398G	1.1G	377G	1%	/data

/data が正常にマウントされている事が確認出来ました。

フローティング IP アドレスの変更はありません。

```
[root@xmon-cl1 ~]# /sbin/ifconfig eth0:0
```

```
eth0:0    Link encap:Ethernet  HWaddr 00:0C:29:AB:3A:25
          inet addr:192.168.24.133  Bcast:192.168.24.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

xmon-cl2 でも確認しておきます。

```
[root@xmon-cl2 ~]# crm_mon -n1
```

=====

～中略～

Current DC: xmon-cl1 - partition with quorum

～中略～

=====

Node xmon-cl1: online

```
res_ln-x-mon    (lsb:ln-x-mon) Started
res_snmpptt     (lsb:snmpptt) Started
res_nsca        (lsb:nsca) Started
res_vip (ocf::heartbeat:IPaddr2) Started
```

```
res_filesystem (ocf::heartbeat:Filesystem) Started
res_httpd      (lsb:httpd) Started
res_postgresql (lsb:postgresql) Started
res_drbd:0     (ocf::linbit:drbd) Master
res_x-mon      (lsb:x-mon) Started
res_snmptrapd  (lsb:snmptrapd) Started
```

Node xmon-cl2: online

```
res_drbd:1     (ocf::linbit:drbd) Started
```

xmon-cl1 と同じ内容が表示され、xmon-cl1 がマスターで各リソースが動作している事が確認出来ました。

データ領域の確認をしてみます。

```
[root@xmon-cl2 ~]# service drbd status
```

～中略～

```
m:res  cs          ro          ds          p mounted fstype
0:r0   Connected  Secondary/Primary UpToDate/UpToDate C
```

自身がセカンダリになり、マウントしていない事が確認出来ます。

フローティング IP アドレスが割り当てられていない事も確認しましょう。

```
[root@xmon-cl2 ~]# /sbin/ifconfig eth0:0
```

```
eth0:0    Link encap:Ethernet  HWaddr 00:0C:29:0F:B3:D9
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

システムからも /data をマウントしていない事を確認します。

```
[root@xmon-cl2 ~]# df -Th
```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
/dev/sda5	ext4	9.7G	529M	8.7G	6%	/
tmpfs	tmpfs	3.9G	17M	3.9G	1%	/dev/shm
/dev/sda1	ext4	485M	49M	412M	11%	/boot
/dev/sda6	ext4	9.7G	1.1G	8.2G	12%	/usr
/dev/sda2	ext4	20G	333M	18G	2%	/var

7.1.3.7 /etc/opt/nrpe/nrpe.cfg ファイルの変更

/etc/opt/nrpe/nrpe.cfg ファイルの nrpe 許可設定を変更します。

■対象ファイル

/etc/opt/nrpe/nrpe.cfg

■対象機器

xmon-cl1, xmon-cl2 ともに変更を実施します。変更箇所は共通です。

バックアップを取得します。

```
[root@xmon-cl1 ~]# cp -vip /etc/opt/nrpe/nrpe.cfg ~/backup/  
`/etc/opt/nrpe/nrpe.cfg' -> `/root/backup/nrpe.cfg'
```

xmon-cl1, xmon-cl2 ともに設定ファイルの書き換えを実施してください。

```
[root@xmon-cl1 ~]# vi /etc/opt/nrpe/nrpe.cfg
```

■編集前

```
allowed_hosts=127.0.0.1, 10.0.0.1, 10.0.0.2
```

■編集後

```
allowed_hosts=127.0.0.1, 172.16.16.1, 172.16.16.2
```

反映のため nrpe を再起動します。

```
[root@xmon-cl1 ~]# /etc/init.d/nrpe restart
```

xmon-cl1、xmon-cl2 ともに再起動を実施してください。

7.1.3.8 監視設定変更

X-MON 管理画面より xmon-cl1、xmon-cl2 の IP アドレスを変更してください。

ホスト名称	<input type="text" value="xmon-cl1"/>
種別	<input type="text" value="物理サーバ"/>
ロゴイメージ	<input type="text" value="不明"/>
?	
IPアドレス/FQDN	<input type="text" value="172.16.16.1"/>
MACアドレス	

変更後 X-MON の再起動を実施してください。

以上でデータ同期用 IP アドレスの変更は完了です

7.1.3.9 同じネットワーク内で IP アドレスが変更する場合

同じネットワーク内での変更の場合は物理 IP アドレスの変更後、corosync の設定変更は必要ありません。そのため、手順としては

- | | |
|---------|--------------------------------|
| 7.1.3.1 | corosync を停止 |
| 7.1.3.2 | 物理 IP アドレスの変更 |
| 7.1.3.3 | /etc/hosts ファイルの変更 |
| 7.1.3.4 | DRBD の設定変更 |
| 7.1.3.6 | corosync を起動 |
| 7.1.3.7 | /etc/opt/nrpe/nrpe.cfg ファイルの変更 |
| 7.1.3.8 | 監視設定変更 |

となります。

7.2 snmptrapd のコミュニティ名の変更手順

X-MON では snmptrapd のデフォルトのコミュニティ名は public となっています。
変更が必要な場合の手順を記載します。

7.2.1 X-MON のマスターの確認

通常、X-MON のマスターは xmon-cl1 で動作しています。
本手順に関してもマスターは xmon-cl1 で動作している事を前提として記載しています。
マスターに関しては X-MON の監視画面で確認出来ます。クラスタが切り替えを行っている環境（xmon-cl2 がマスター）の場合は、次章以降のマスターの部分を読み替えて作業を実施してください。

図 X-MON

X-MON (X-MON)	Maser Node	正常(OK)	2012-11-21 16:54:00	9日と09時間48分 58秒	1/3	Master node: xmon-cl1
	PING	正常(OK)	2012-11-21 16:53:59	9日と09時間47分 29秒	1/3	PING OK - Packet loss = 0%, RTA = 0.03 ms

Master node: xmon-cl1

「Master node:xmon-cl1」となっています。

7.2.2 変更手順

■対象ファイル

/etc/snmp/snmptrapd.conf

■対象機器

```
[root@xmon-cl1 ~]# ls -l /etc/snmp/snmptrapd.conf
-rw----- 1 root root 234 Jul 13 08:10 /etc/snmp/snmptrapd.conf
```

リンクが表示されない場合は「xmon-cl1, xmon-cl2」ともに作業が必要です。

```
[root@xmon-cl1 ~]# ls -l /etc/snmp/snmptrapd.conf
lrwxrwxrwx 1 root root 25 Jul 18 11:36 /etc/snmp/snmptrapd.conf ->
/data/snmp/snmptrapd.conf
```

/data/snmp/snmptrapd.conf にリンクが貼られている場合は、「xmon-cl1」のみ作業が必要です。

バックアップ用ディレクトリを作成し、バックアップを取得します。

```
[root@xmon-cl1 ~]# mkdir ~/backup
[root@xmon-cl1 ~]# cp -vip /etc/snmp/snmptrapd.conf ~/backup/
```

```
`/etc/snmp/snmptrapd.conf' -> `snmptrapd.conf'
```

次の行の「public」部分を変更されたいコミュニティ名に修正してください。

例：コミュニティ名を clusxmon に変更

```
[root@xmon-cl1 ~]# vi /etc/snmp/snmptrapd.conf
```

■編集前

```
authCommunity log, execute, net public
```

■編集後

```
authCommunity log, execute, net clusxmon
```

また、複数のコミュニティ名を設定したい場合は

上記の行をコミュニティ名分、記載してください。

例) public,public1,public2 と 3 つのコミュニティ名を設定する場合

■編集前

```
authCommunity log, execute, net public
```

■編集後

```
authCommunity log, execute, net public
authCommunity log, execute, net public1
authCommunity log, execute, net public2
```

上記修正後、X-MON が稼働しているマスター側 (xmon-cl1) で次のように snmptrapd サービスを再起動します。

もう一台のスレーブ (xmon-cl2) については必要に応じてファイルの修正のみ実施いただき、snmptrapd の再起動は不要です。

```
[root@xmon-cl1 ~]# service snmptrapd restart
```

```
Stopping snmptrapd: [ OK ]
Starting snmptrapd: [ OK ]
```

以上で snmptrapd のコミュニティ名の変更は完了です。