

# Paxos iSCSI (評価版)

## 取扱説明書

---

Paxos Storage Service(PSS)を backing store にした iSCSI target

1. 0. 2

**2011/08/17**

※最近の disk 利用は、iSCSI によるものが多くなりました。Paxos Storage Service のファイルを backing store にした iSCSI target を作成しました。

株式会社トライテック

## ■ 改版履歴

版	修正日	作成者	変更点
1.0.1 版	2011/07/27	久保 正業	新規作成
1.0.2 版	2011/08/17	久保 正業	RPM を追加

## ■ 目次

1. 概要.....	2
2. 実現方式.....	3
3. 利用形態例.....	4
4. 使用方法.....	6
4.1. マシン構成.....	6
4.2. パッケージの種類.....	7
4.3. Paxos iSCSI のインストール.....	9
4.3.1. PSS サーバ構成の決定と設定ファイルの配布.....	9
4.3.2. PSS サーバのインストール.....	9
4.3.3. PSS サーバ制御のインストール.....	9
4.3.4. iSCSI ターゲットのインストール.....	9
4.3.5. PSS サーバの初期化と起動.....	9
4.3.6. PSS サーバ制御の初期化と制御コマンド.....	9
4.3.7. PSS サーバの終了と再起動.....	11
4.3.8. PSS サーバのログ.....	12
4.4. PSS 上にボリューム作成.....	12
4.5. iSCSI target の起動と停止.....	12
4.6. 状態の表示.....	12
4.7. iSCSI initiator からの接続 (iSCSI target の利用).....	13
4.7.1. Linux の場合.....	13
4.7.2. Windows の場合.....	14

## 1. 概要

Paxos Storage Service (PSS)の高可用性のストレージ機能を使用するひとつのインターフェースとして iSCSI インターフェースの提供形態を用意しました。この iSCSI インターフェースは、以下の特徴があります。

- ・ iSCSI I/F で、Paxos サーバの存在を意識せずに、ユーザ選択の自由なファイルシステムで使用できます。
- ・ ディスクとして高価な RAID の冗長性の確保以上のデータの信頼性の向上が図れます。
- ・ サーバを分散すれば、災害にも強く、自然にデータのバックアップを実現できます。

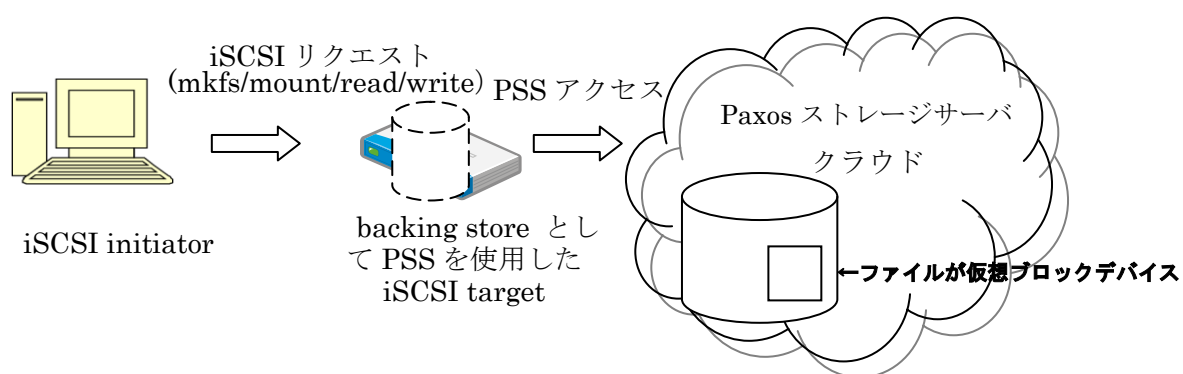


図 1 backing store として PSS を使用した iSCSI target

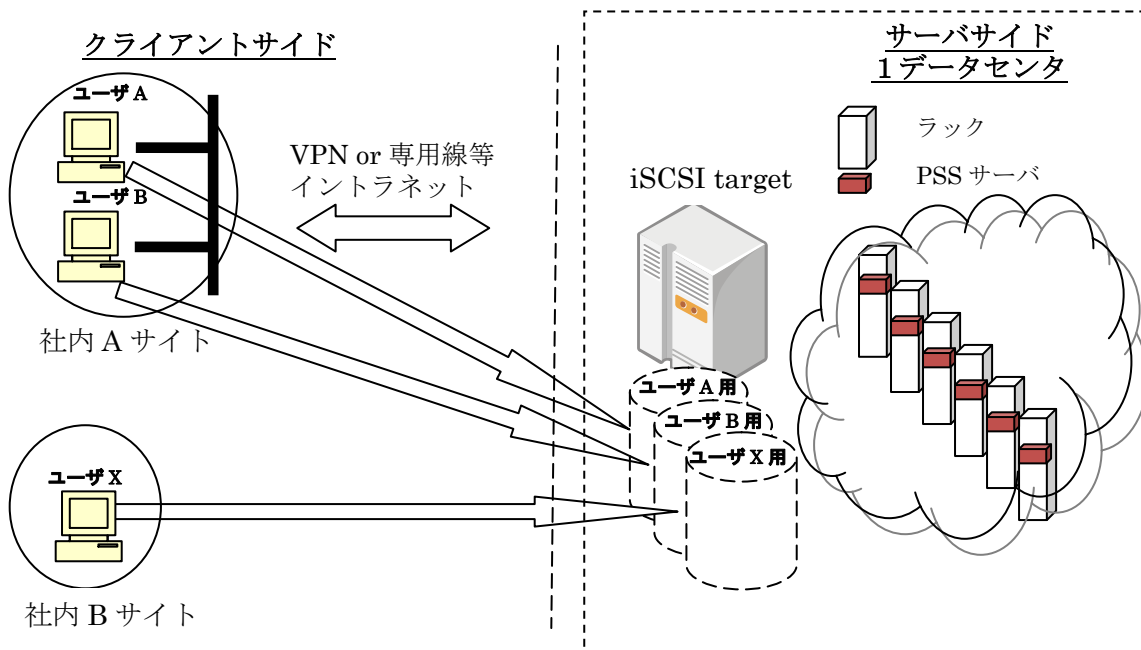
## 2. 実現方式

提供する iSCSI ターゲットは、TGT version 1.0.17 をもとに **backing store** のアクセスを通常ファイルから PSS へのアクセスに変更したものです。「4. 使用方法」にしたがって、インストール/環境設定/デーモン起動/PSS タイプの LUN 作成していただければ、ネットワークに公開可能です。

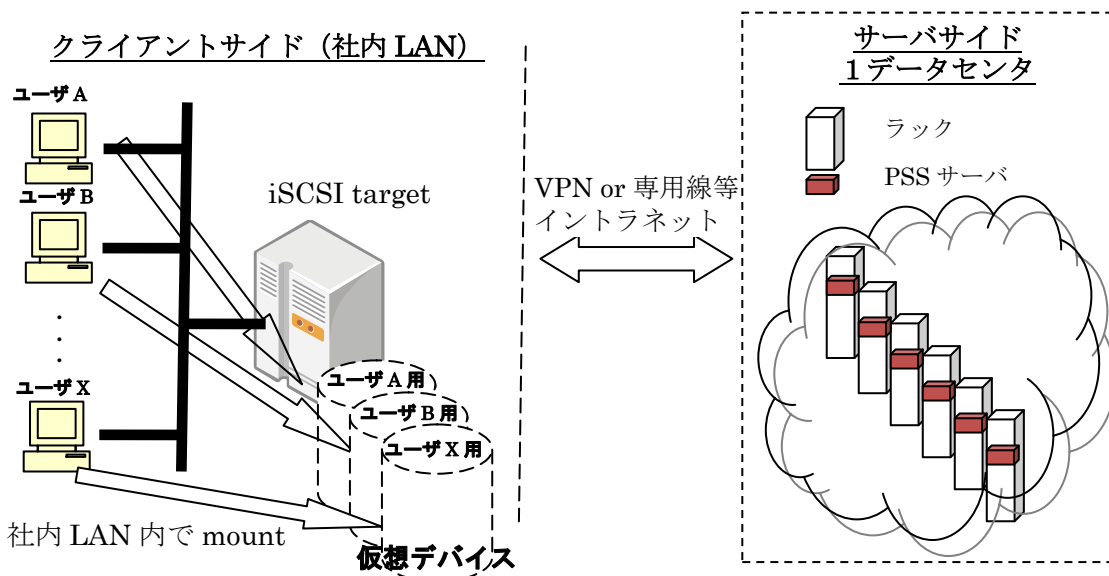
PSS (Paxos Storage Service) は、iSCSI target のデーモン起動前に（正確には、LU(ロジカルユニット)作成前に) デーモンプログラムからアクセスできるようにサーバを配置する必要があります。その方法は、「4. 使用方法」に詳しく記述してあります。

### 3. 利用形態例

- (1) iSCSI target と PSS サーバをサーバサイドに置いて、場所が分かれていても社内の別の場所からでもアクセスできます。



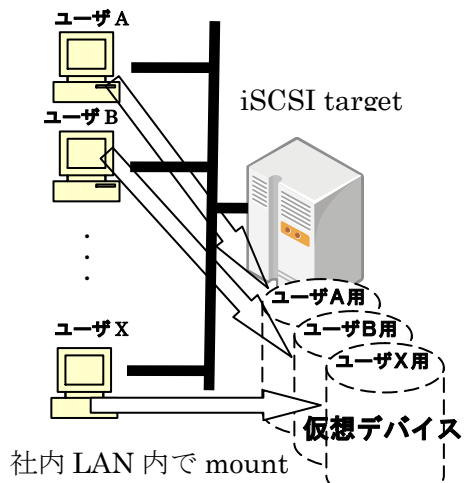
- (2) クライアントサイドに iSCSI target を置いて PSS サーバをデータセンタに置いて、データと処理を分散する。極端には,initiator/target 同マシンの構成もできます。



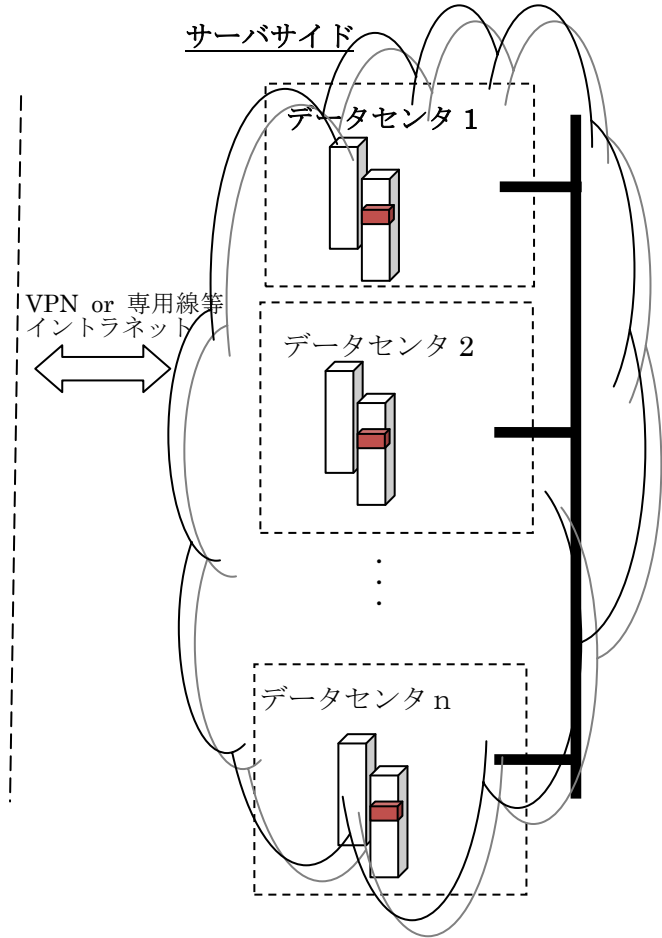
(3) PSS サーバ分散配置

複数のデータセンターを使って、PSS サーバを分散。危機管理を強化する。

クライアントサイド



サーバサイド



## 4. 使用方法

### 4.1. マシン構成

- (1) PSS サーバマシン (VM でも可) 3 台以上 8 台まで
- (2) iSCSI ターゲット (VM でも可) 1 台 (※PSS サーバ制御マシンと同一でもよい)
- (3) PSS サーバ制御 (VM でも可) 1 台 (※iSCSI ターゲットマシンと同一でもよい)

OS は、Ubuntu11.04(amd64) desktop 版か CentOS5.6(final)でお願いします。

それぞれのマシンには、共通のユーザと SSH (クライアントとサーバ) を入れてください。また *CentOS* の場合は、ファイアウォールの設定に気をつけてください。udp は全てのポートでの通信を許可 (PSS サーバマシンと PSS サーバ制御マシン) し、tcp は、PSS サーバは、使用するポート (4010~4017) をそれぞれあけ、iSCSI ターゲットは、3206 をあけておいてください。以下のような行を iptables の RH-Firewall chain に追加してください。

udp のフィルタ追加

```
-A RH-Firewall-1-INPUT -p udp -m udp -j ACCEPT ←これは、dport 指定を外す意味です。
```

tcp のフィルタ追加

```
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 4010:4017 -j ACCEPT
```

←各 PSS サーバ1つのポートでもよい

```
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 3206 -j ACCEPT ←iSCSI ターゲットの場合
```

また、(2) iSCSI ターゲットマシンには、libconfig-general-perl

(Ubuntu) (perl-Config-General (CentOS) ) パッケージをインス

トールしててください。

以下に、マシン構成の例を示します。

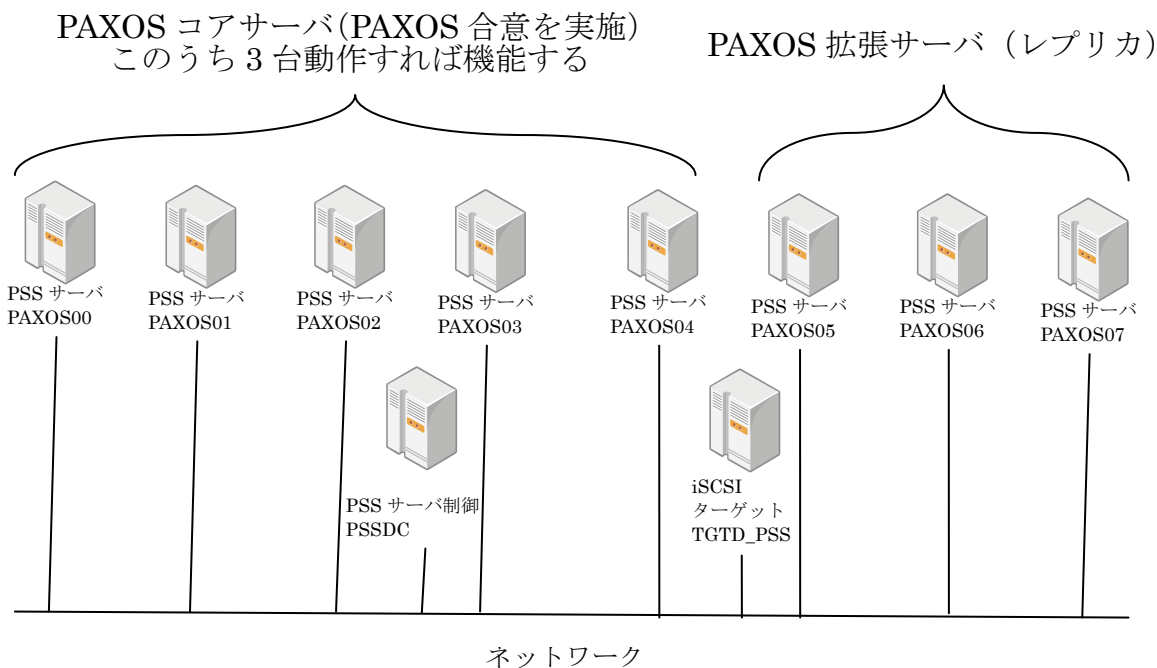


図 5 マシン構成図

#### 4.2. パッケージの種類

以下のパッケージと設定ファイル (例) があります。

- (1) PSS サーバ  
 ファイル名 : pssd.deb (Ubuntu)、pss-server-1.0.1-2.x86\_64.rpm(CentOS)  
 パッケージ名 : pss-server
- (2) PSS サーバ制御  
 ファイル名 : pssdc.deb(Ubuntu)、  
                   pss-server-controller-1.0.1-2.x86\_64.rpm(CentOS)  
 パッケージ名 : pss-server-controller
- (3) iSCSI ターゲット  
 ファイル名 : tgtd.pss.deb(Ubuntu)、  
                   tgt-pss-target-1.0.1-2.x86\_64.rpm(CentOS)  
 パッケージ名 : tgt-pss-target
- (4) server.conf  
 /etc/pss ディレクトリにおいて、PSS サーバ構成を記述したものです。この設定  
 ファイルは、PSS サーバマシン、PSS サーバ制御マシン、iSCSI ターゲットマ  
 シンの全てに必要です。最初に PSS サーバの構成を決めた時に全マシンは配布

しておいてください。図5と同じPSSサーバ構成で、サーバのDNS(または/etc/hostsでの指定した)名をpaxos00~paxos07としたとき以下のようになります。

```
# id server path
0 paxos00 /pss
1 paxos01 /pss
2 paxos02 /pss
3 paxos03 /pss
4 paxos04 /pss
5 paxos05 /pss
6 paxos06 /pss
7 paxos07 /pss
```

1列目のidのところには、0~7をかきます。サーバ番号で役割は、0~4まではコアサーバといって、実際のPAXOS合意を行うサーバで、どれも平等で役割の違いはありません。5~7は、PAXOS合意には参加しませんが、リプリケーションのように0~4までのコアサーバと同じデータを保持します。

2列目のserverのところは、サーバ名またはIPアドレスを記述します。

3列目のpathのところは、データの保持場所でiSCSIで見えるボリューム(複数可能)の置き場所(制御データおよび内部の一時データも含む)です。確保したサイズだけのデータを作成できます。複数パス(デバイス)は使用できません。

**※ デバイスは指定できません。必ずフォーマットしてマウントしておいてください。**

最低で0~4のうちの3台があればPSSは動作しますが、server.confは、8レコード必要で、2列目のserverのところは、DNS又は/etc/hostsで名前解決ができるか、IPアドレスでなければなりません。

それぞれのサーバは、udpポート4000+id,tcpポート4010+idを使用します。

#### (5) targets.conf

iSCSIターゲットマシンの/etc/tgtに置いて、ターゲット名、対応するPSSファイル名、アクセス可能ユーザ名/パスワード等を指定します。例では、以下のようになります。

```
<target iqn.2001-03.jp.tritech:tgtd-pss>
  <backing-store vol001>
    bs-type pss
    lun 1
  </backing-store>
  incominguser iscsi_pss iscsi_pss000
</target>
```

target タグにターゲット名(IQN)を指定します。複数のターゲットを作成する場合は、<target>~</target>を繰り返します。target タグ中のbacking-store タグで、PSS ファイル名を指定します。このバックングストアは、createISCSIVol コマンドで作成します(後述)。backing-store タグ中にbs-type とlunを指定します。bs-typeは、pssを必ず指定します。これによって、Paxos iSCSIを使用することになります。lunは、論理ユニット番号で1つのターゲット

トで複数のバックングストアを使用する場合に区別するための番号です。ターゲットでユニークにした 1 以上の番号をバックングストアに与えます。

`incominguser` は、そのターゲットにアクセスできるユーザをユーザ名とパスワードを指定して制限するものです。

※ `initiator-address` を指定してログイン可能なイニシエータの制限もできます。

### 4.3. Paxos iSCSI のインストール

#### 4.3.1. PSS サーバ構成の決定と設定ファイルの配布

マシン構成を決めます。たとえば、PSS サーバの DNS 名を `paxos00~paxos07` をとし、PSS サーバ制御マシンの DNS 名を `pssdc`、iSCSI ターゲットマシンの DNS 名を `tgt_d_pss` とした場合に、PSS サーバ構成 (PSS サーバのデータパスを含めて) を `server.conf` に記述して(4.2 パッケージの種類 (4) 参照)、全マシンに配布して `/etc/pss` に置きます。

#### 4.3.2. PSS サーバのインストール

`paxos00~paxos07` に PSS サーバパッケージをインストールします。

```
sudo dpkg -i pssd.deb (Ubuntu)
```

```
rpm -ihv --force pss-server-1.0.1-2.x86_64.rpm(CentOS)
```

※ CentOS は root で実行する。以下同様。

#### 4.3.3. PSS サーバ制御のインストール

`pssdc` に PSS サーバ制御パッケージをインストールします。

```
sudo dpkg -i pssdc.deb (Ubuntu)
```

```
rpm -ihv pss-server-controller-1.0.1-2.x86_64.rpm(CentOS)
```

#### 4.3.4. iSCSI ターゲットのインストール

`tgt_d_pss` に iSCSI ターゲットパッケージをインストールします。

```
sudo dpkg -i tgt_d.pss.deb
```

```
rpm -ihv --force pss-pss-target-1.0.1-2.x86_64.rpm(CentOS)
```

#### 4.3.5. PSS サーバの初期化と起動

`paxos00~paxos07` で、初期化してサーバをスタートします。

Ubuntu の場合 :

```
sudo pss_delper.sh
```

```
sudo /etc/init.d/pssd start
```

CentOS の場合(root で行う) :

```
pss_delper.sh
```

```
/etc/init.d/pssd start
```

#### 4.3.6. PSS サーバ制御の初期化と制御コマンド

`pssdc` で、PSS サーバへのパスワードなしログインを可能にするため `ssh` のパス

ワードなしキーを作成します。(CentOS の場合でも root でない各マシン共通ユーザで行ってください)

```
ssh-keygen
```

Enter file in which to save the key (/home/kubo/.ssh/id\_rsa):

とでたら、入力なしでEnterを入力

Enter passphrase (empty for no passphrase):

とでたら、入力なしでEnterを入力

Enter same passphrase again:

とでたら、入力なしでEnterを入力

これで、~/.ssh/id\_rsa.pubができます。そして、id\_rsa.pubをpaxos00~paxos07に配布して~/.ssh/authorized\_keysに設定。

```
pss_set_ssh.sh
```

これで、各コマンドがスムーズに実行可能になります。

(1) サーバ生死確認

```
pss_lookup-server.sh
```

以下のように表示 (0~7にないものがあれば死んでいます)

```
9326 ?      Sl      2:26 /pss/PFSserver 0 /pss/PFS-0
13005 ?     Sl      3:47 /pss/PFSserver 1 /pss/PFS-1
5133 ?      Sl      2:10 /pss/PFSserver 3 /pss/PFS-3
12760 ?     Sl      3:05 /pss/PFSserver 4 /pss/PFS-4
5287 ?      Sl      1:58 /pss/PFSserver 6 /pss/PFS-6
 9018 ?     Sl      0:29 /pss/PFSserver 7 /pss/PFS-7
```

(2) PSSのファイル操作コマンド

```
PFSClient readdir / ← パス名指定
```

```
PFSClient stat /vol001 ← ファイル状態取得
```

```
PFSClient read /vol001/00/00/01 file1 ← ファイル取得
```

```
etc. ....
```

(3) PSSサーバシャットダウン

```
PSSShutdown 0 ← id
```

(4) PSSサーバ状態取得

```
PFSProbe 0 accept ← 接続クライアント情報表示
```

```
PFSProbe 1 outbound ← 保持中のアウトバウンドデータ
```

```
PFSProbe 2 fd ← オープン中のファイル情報表示
```

etc....

(5) Paxos情報取得

PSSAdmin 0 control ← サーバ間の接続情報

PSSAdmin 1 paxos [ページ] ← PAXOS合意の表示

etc....

(6) ポート (tcp/udp) 情報取得

サーバが突然死んだ場合に、すぐ立ち上がらないことがあるます。固定ポートをバインドできない場合です。そのために各サーバのポート情報を表示するコマンドを用意しました。

pss\_lookup\_port.sh

以下のように表示

```
tcp      0      0 10.1.1.182:4010      0.0.0.0:*      LISTEN
tcp      0      0 10.1.1.182:4010      10.1.1.140:60286 ESTABLISHED
tcp      0      0 10.1.1.182:4010      10.1.1.140:60277 ESTABLISHED
tcp      0      0 10.1.1.182:4010      10.1.1.140:60279 ESTABLISHED
. . . . .
. . . . .
udp      0      0 10.1.1.182:4000      0.0.0.0:*
```

#### 4.3.7. PSS サーバの終了と再起動

PSS サーバを終了させるためには、

`sudo /etc/init.d stop (Ubuntu)`

`/etc/init.d stop (CentOS:root で実施)`

です。

初期起動以外のサーバの起動は、3つの場合に分けられます。

1) 最初の1台目のサーバの立上

全てのサーバが落ちてしまった（あるいは落とされた）場合に、立上げ手順が必要になり、まず信頼できるデータを持ったサーバを id が 0~4 までのコアサーバのうちから1つ選んで最初に立上げます。

`sudo /etc/init.d/pssd restart (Ubuntu)`

`/etc/init.d/pssd restart (CentOS:root で実施)`

2) 3台目までのコアサーバの立上

コアサーバ (id が 0~4 までのサーバ) が 3台立ち上がっていない (1~2台立ち上がっている) 場合は、データを参照するサーバ ID を指定して立上ります。(指定サーバのデータを参照してそれに合わせてから動作する)

以下のような立上になります。

Ubuntu の場合 :

```
sudo /etc/init.d/pssd restart autonomic id ← 「autonomic」は、そのまま文字列”autonomic”と入力。id は、先に立ちあがっているデータを参照するサーバの ID
```

CentOS の場合 (root で実施) :

```
/etc/init.d/pssd restart autonomic id
```

3) コアサーバ過半数 (3 台以上) 立ち上がっている場合

コアサーバ過半数立上っている場合は、データを参照するサーバ ID を指定せず、PAXOS 合意で決まったマスタサーバ (指定なし) を参照して立上げできます。

```
sudo /etc/init.d/pssd restart autonomic (Ubuntu)
```

```
/etc/init.d/pssd restart autonomic (CentOS:root で実施)
```

#### 4.3.8. PSS サーバのログ

PSS サーバのログは、シスログ (/var/log/syslog (Ubuntu)、/var/log/messages(CentOS)) にでます。タグは pssd\_0 等 pssd\_サーバ ID です。ログである程度の状況を把握できます。

#### 4.4. PSS 上にボリューム作成

PSS 上に作成するターゲットが使用するボリューム (server.conf で指定した path とは違います) を作成します。任意の名前でよいですが、target.conf に書いた名前のボリューム名 (バックリングストア名) のボリュームを作成します。

```
createISCSIVol volName num{G|T}
```

ex. createISCSIVol vol001 1T <-1T バイトのボリュームをボリューム名「vol001」として作成します。

※ creatISCSIVol コマンドは、/sbin にインストールされます。

#### 4.5. iSCSI target の起動と停止

```
起動 : sudo /etc/inid.d/tgtd.pss start (Ubuntu)
```

```
      /etc/inid.d/tgtd.pss start (CentOS:root で実施)
```

```
停止 : sudo /etc/inid.d/tgtd.pss stop
```

```
      /etc/inid.d/tgtd.pss stop (CentOS:root で実施)
```

#### 4.6. 状態の表示

iSCSI target の状態やアカウントの状態は以下のようにして表示します。

Ubuntu の場合 :

```
sudo tgtadm --lld iscsi --op show --mode target ←iSCSI target
```

(あるいは、 sudo tgt-admin -s)

```
sudo tgtadm --lld iscsi --op show --mode accunt ←アカウント
```

CentOS の場合(root で実施) :

```
tgtadm --lld iscsi --op show --mode target
```

(あるいは、tgt-admin -s)

```
tgtadm --lld iscsi --op show --mode accunt
```

#### 4.7. iSCSI initiator からの接続 (iSCSI target の利用)

実際のクライアント (iSCSI initiator) は、どのような手続きで Paxos iSCSI を利用できるのかを以下に説明します。

##### 4.7.1. Linux の場合

Linux の場合は、以下の手順で iSCSI target に接続します。

(1) ユーザ/パスワードの設定

/etc/iscsi/iscsid.conf の node.session.auth.username と

node.session.auth.password に接続する iSCSI target に設定したアカウントを書きます。

```
node.session.auth.username = iscsi_pss
```

```
node.session.auth.password = iscsi_pss000
```

(2) iSCSI initiator の起動(open iscsi)

```
sudo /etc/init.d/open_iscsi start(Ubuntu)
```

/etc/init.d/open\_iscsi start(CentOS:root で実施)

(3) iSCSI target の探索

```
sudo iscsiadm --mode=discovery --type=sendtargets -p tgt_d_pss(Ubuntu)
```

```
iscsiadm --mode=discovery --type=sendtargets -p tgt_d_pss(CentOS:root で実施)
```

以下出力例 (iSCSI target 名が列挙されます) です。

```
[tgt_d_pss]:3260,1 iqn.2001-03.jp.tritech: tgt_d_pss
```

```
[tgt_d_pss]:3260,1 iqn.2001-03.jp.tritech: tgt_d_pss-hoge
```

```
[tgt_d_pss]:3260,1 iqn.2001-03.jp.tritech: tgt_d_pss-kubo
```

※ tgt\_d\_pss は、iSCSI target の動作しているホスト名 (IP アドレスでも可)

(4) 接続したい iSCSI target へのログイン

Ubuntu の場合 :

```
sudo iscsiadm --mode=node --login --targetname iqn.2001-03.jp.tritech:tgt_d_pss
```

CentOS の場合 (root で実施) :

```
iscsiadm --mode=node --login --targetname iqn.2001-03.jp.tritech:tgt_d_pss
```

これで新しいデバイスとして,/dev/sd\*が見えます。

(5) フォーマット

最初は未フォーマットなので、必要なら fdisk 等でパーティションを切って、mkfs でファイルシステムを作成します。

(6) マウント

Linux のファイルシステムとしてアクセスするために、`mount` コマンドでどこかのパスにマウントして使用します。

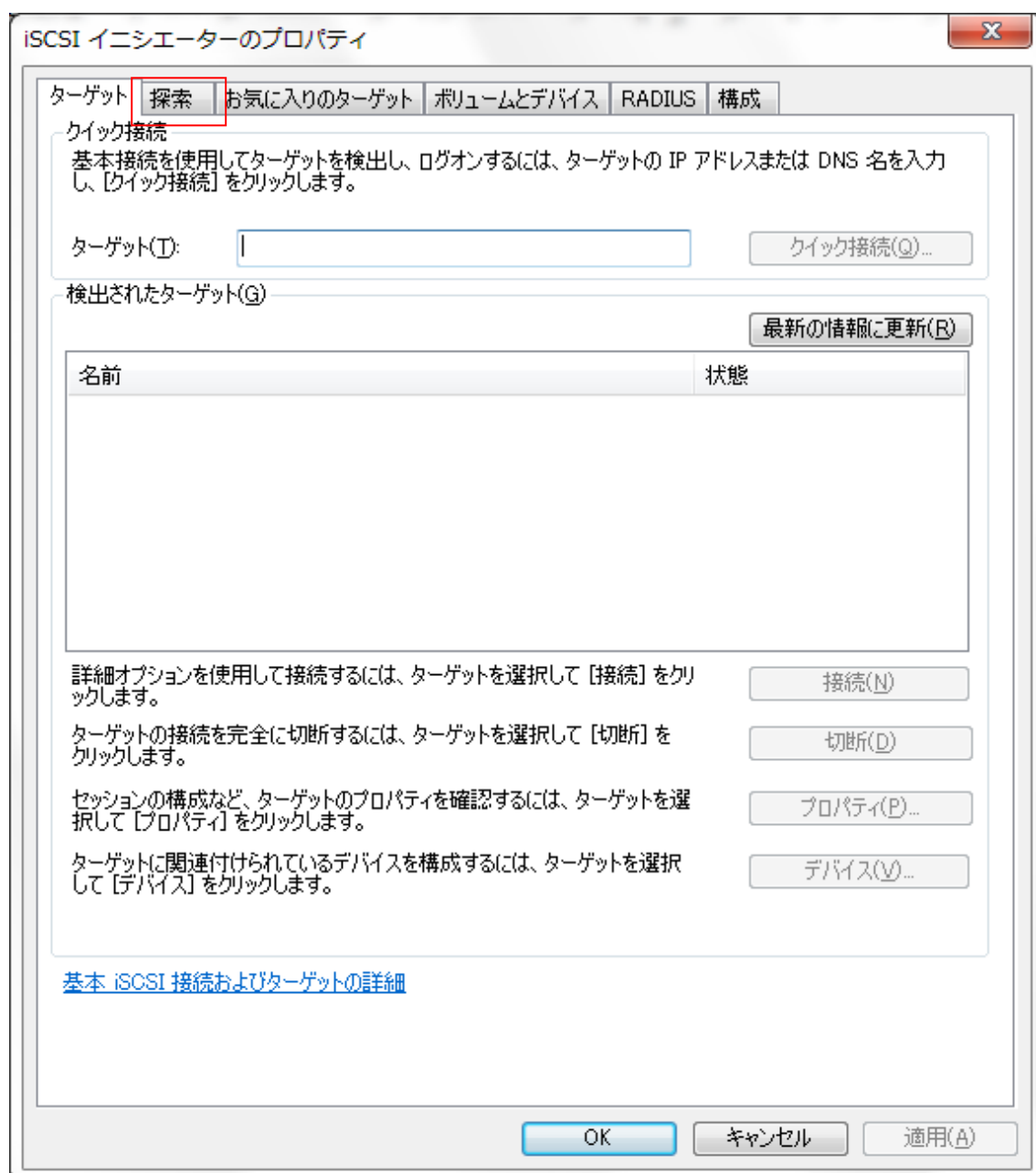
#### 4.7.2. Windows の場合

Windows の場合は、Windows7 の場合について以下の手順を例示します。

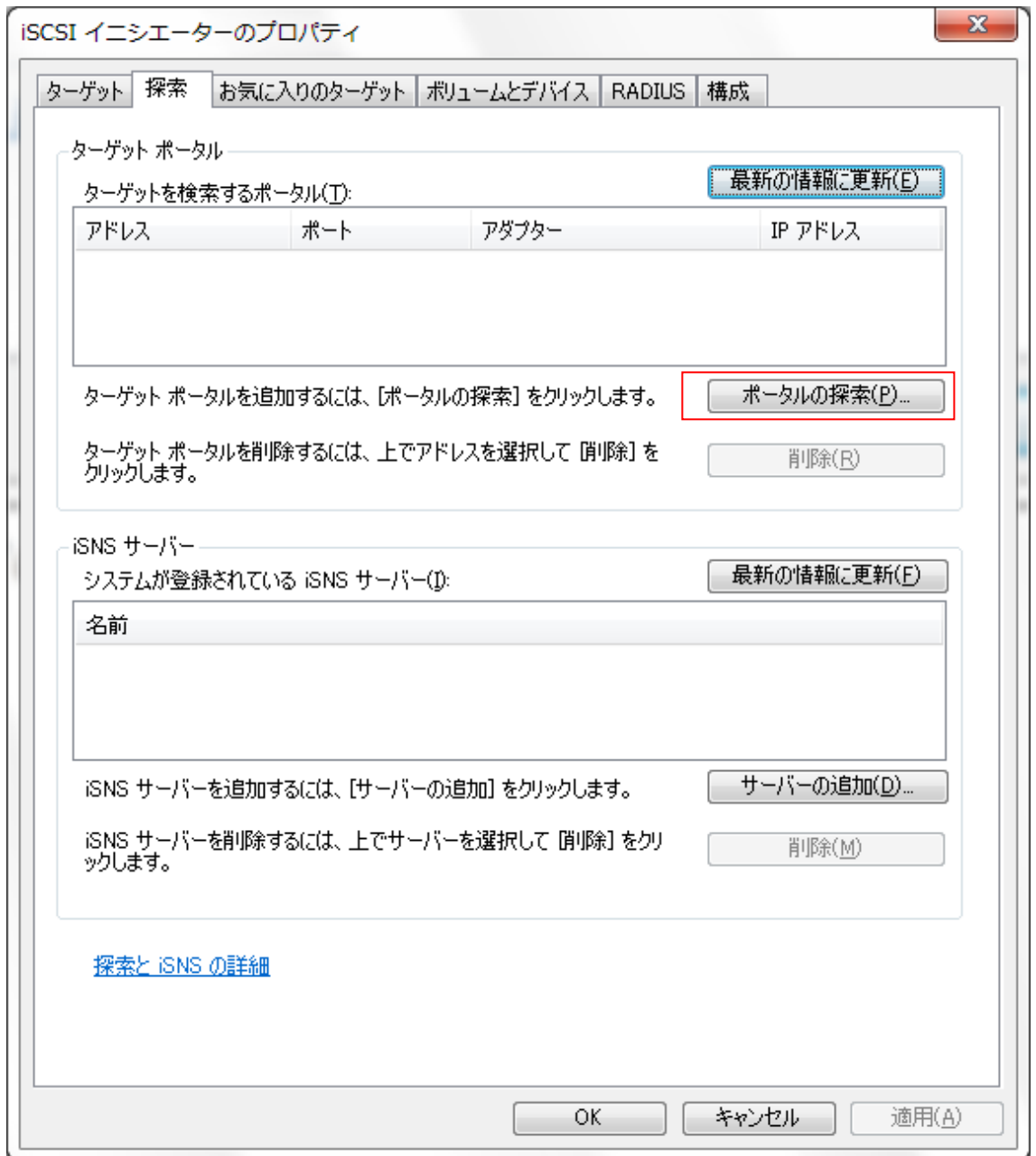
##### (1) iSCSI target の探索

start → コントロールパネル → システムとセキュリティ → 管理ツール  
→ iSCSI イニシエータ

「SCSI イニシエータのプロパティ」ダイアログがでます。

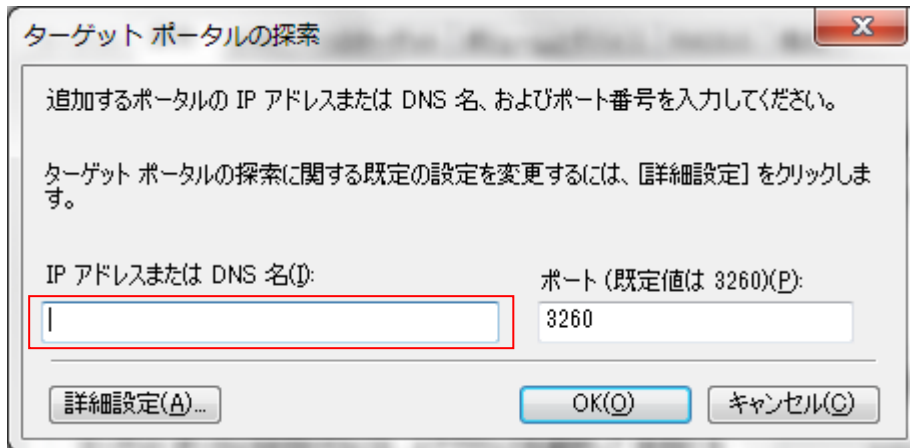


探索タグの画面を出します。

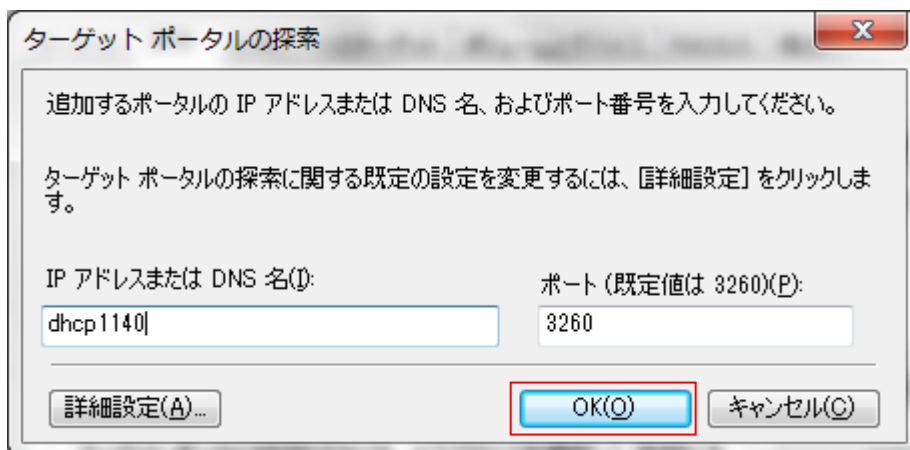


「ポータルの探索」ボタンを押します。

「ターゲット ポータルの探索」ダイアログが表示されます。



iSCSI target の「IP アドレス又は DNS 名」の入力フィールドを入力して「OK」ボタンを押します。

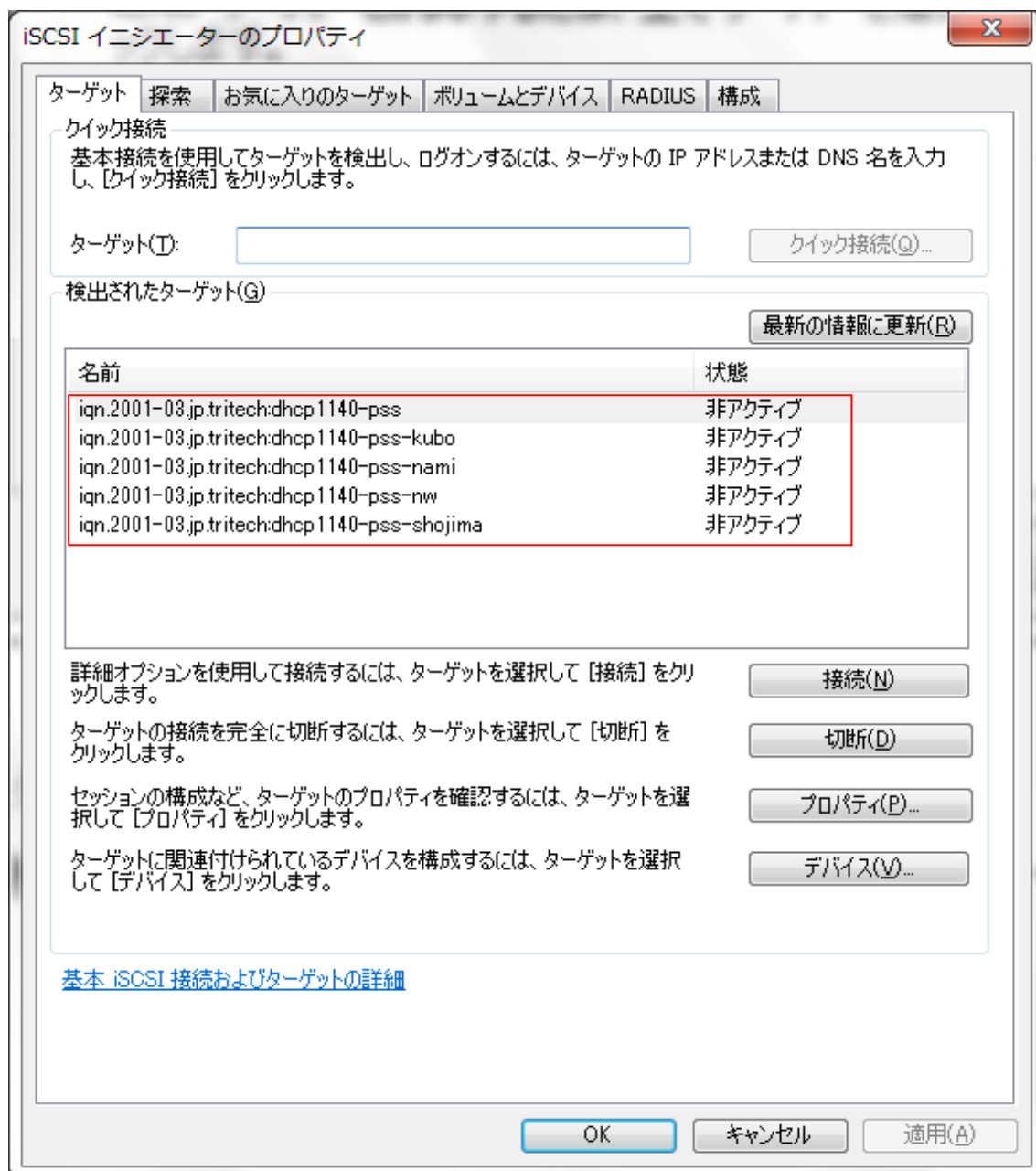


「SCSI イニシエータのプロパティ」ダイアログの「ターゲットを探索するポータル」リストボックスに入力データが表示されます



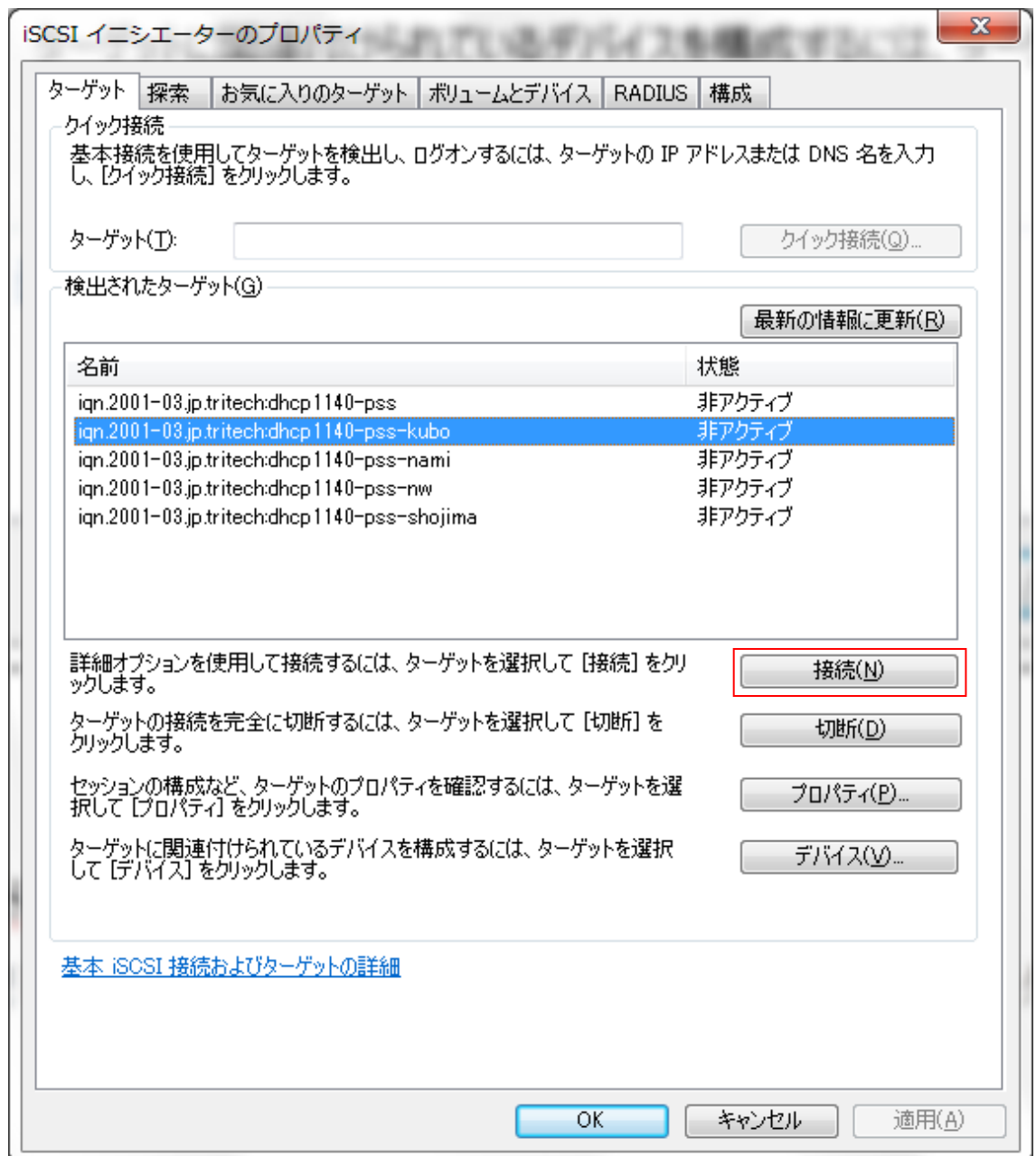
「ターゲット」 タブの画面を出します。

「検出されたターゲット」リストボックスにターゲット名リストが表示されます。

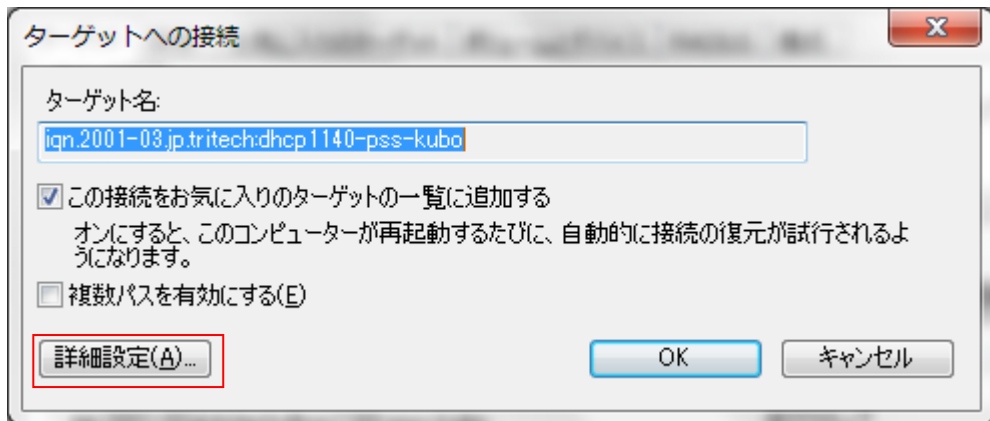


(2) iSCSI target への接続

接続したい「ターゲット」を選択して「接続」ボタンを押します。

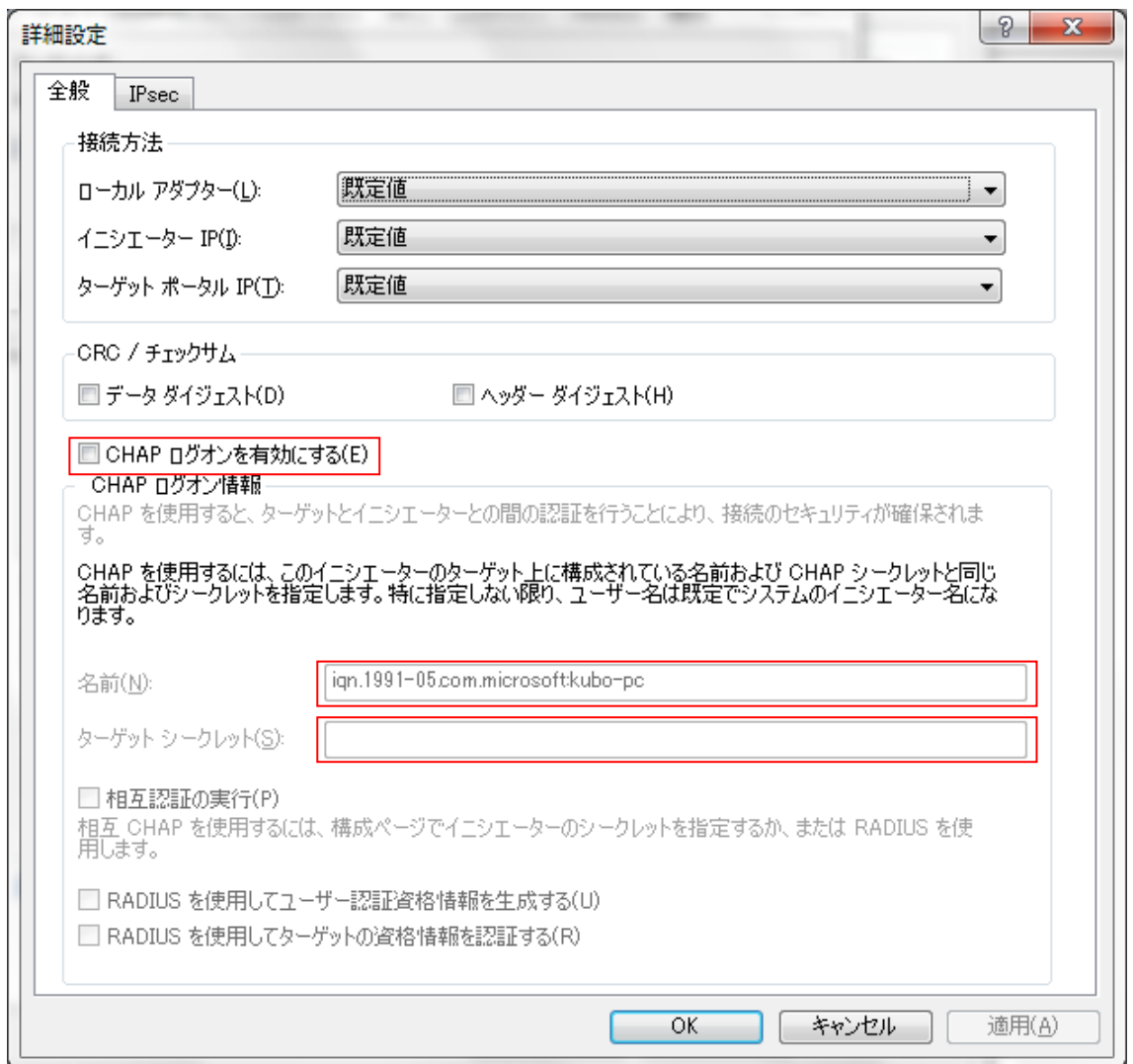


「ターゲットへの接続」ダイアログが出ます。



「詳細設定」 ボタンを押します。

「詳細設定」 ダイアログが出ます。



「CHAP ログインを有効にする」 チェックボックスをチェックし、名前とターゲット シークレットの入力フィールドに 1 ISCSI target で指定したアカウント

のデータを入力して「OK」ボタンを押します。

詳細設定

全般 IPsec

接続方法

ローカル アダプター(L): 既定値

イニシエーター IP(I): 既定値

ターゲット ポータル IP(T): 既定値

CRC / チェックサム

データ ダイジェスト(D)  ヘッダー ダイジェスト(H)

CHAP ログインを有効にする(E)

CHAP ログイン情報

CHAP を使用すると、ターゲットとイニシエーターとの間の認証を行うことにより、接続のセキュリティが確保されます。

CHAP を使用するには、このイニシエーターのターゲット上に構成されている名前および CHAP シークレットと同じ名前およびシークレットを指定します。特に指定しない限り、ユーザー名は既定でシステムのイニシエーター名になります。

名前(N): kubo

ターゲット シークレット(S): ●●●●●●●●●●●●

相互認証の実行(P)

相互 CHAP を使用するには、構成ページでイニシエーターのシークレットを指定するか、または RADIUS を使用します。

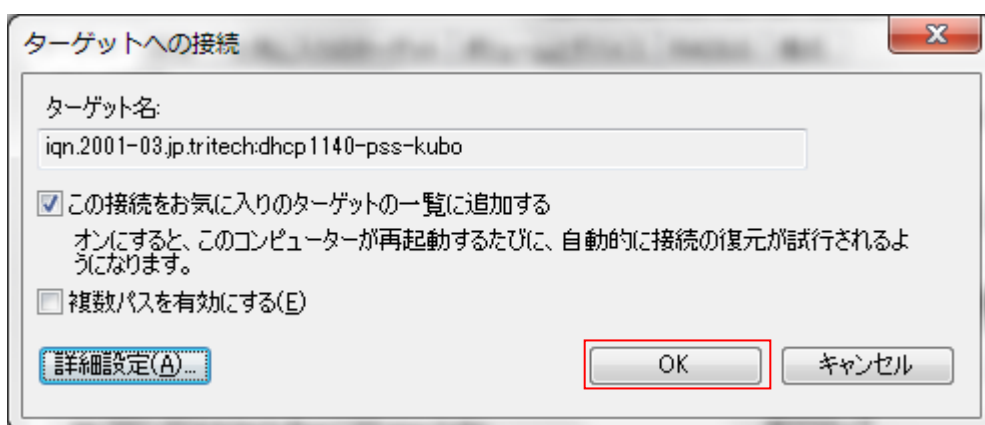
RADIUS を使用してユーザー認証資格情報を生成する(U)

RADIUS を使用してターゲットの資格情報を認証する(R)

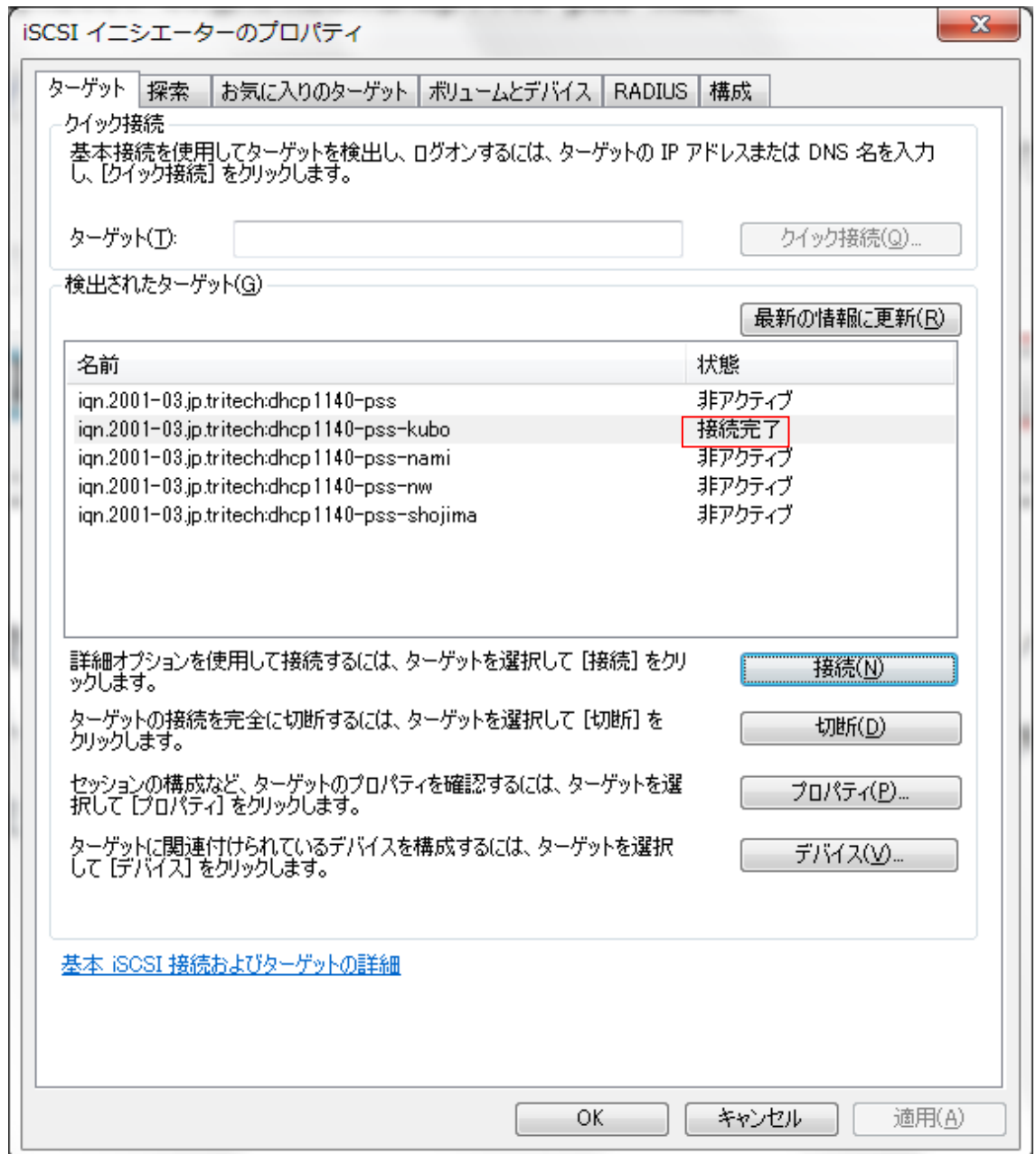
OK キャンセル 適用(A)

※ Windows の場合「ターゲットシークレット」は、12 文字～16 文字と文字数の制限がありますので、iSCSI target での設定に注意してください。

「OK」ボタンを押すと、「ターゲットへの接続」ダイアログ画面に戻りますので、「OK」ボタンを押します。



接続が成功すると、「iSCSI イニシエータのプロパティ」ダイアログの「ターゲット」タブ画面に戻り、対象にした「iSCSI target」の状態が「接続完了」になります。

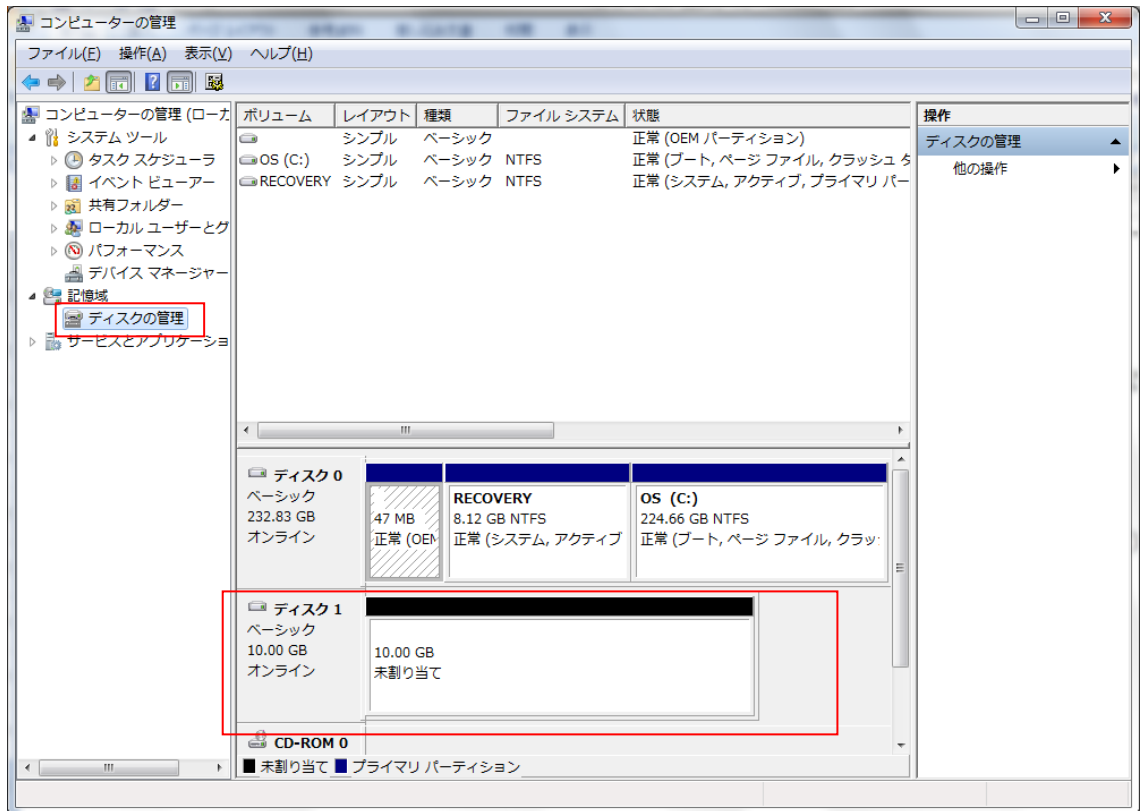


(3) フォーマット

接続ができれば、見えるようになりますが、最初は未フォーマットなので、

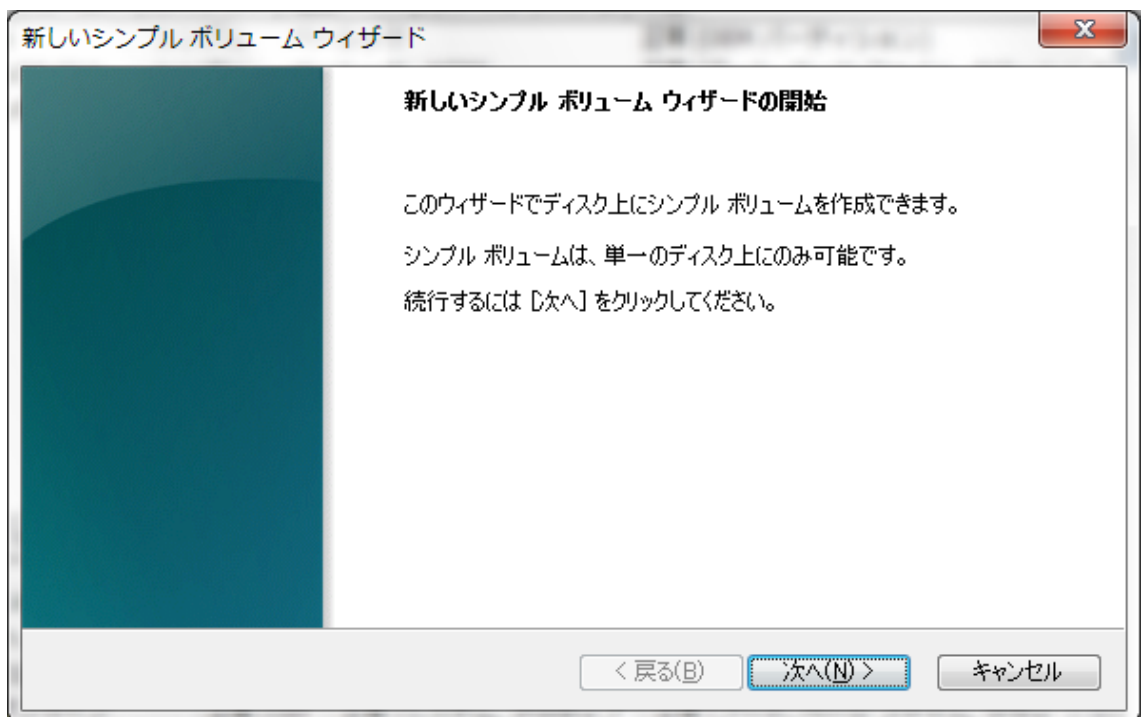
スタート → コントロールパネル → システムとセキュリティ → 管理ツール → コンピュータの管理

「コンピュータの管理」ダイアログ画面の 記憶域 → ディスクの管理

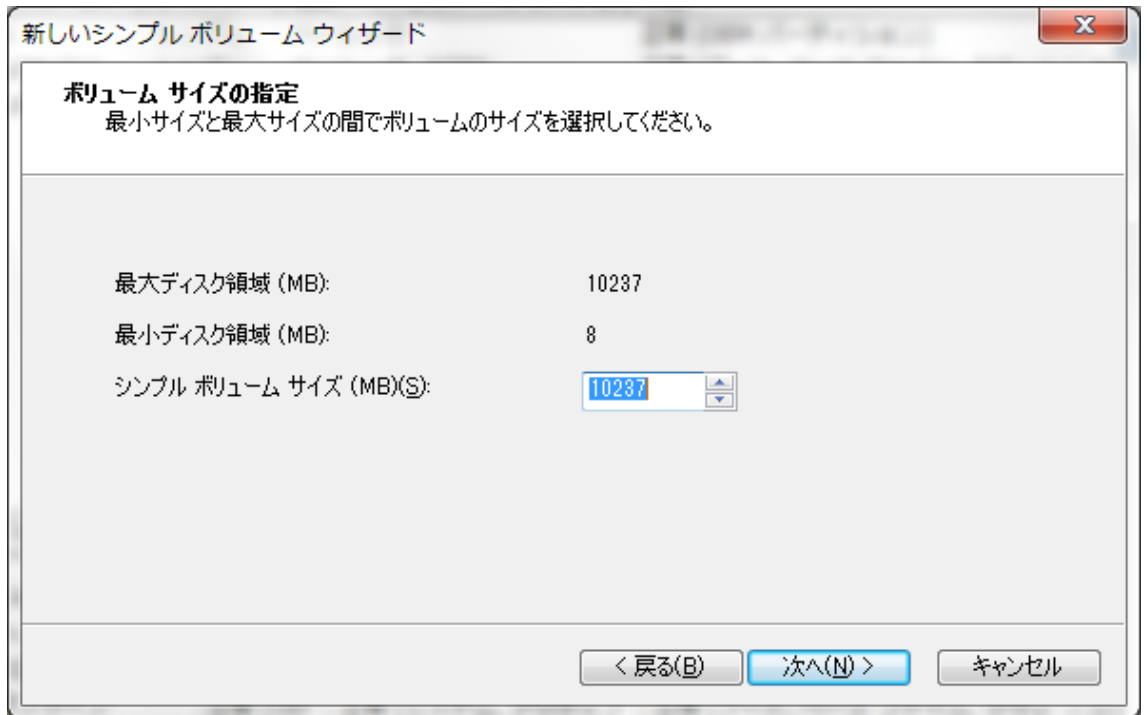


未フォーマットディスクを指定して、「新しいシンプルボリューム」を選択する。

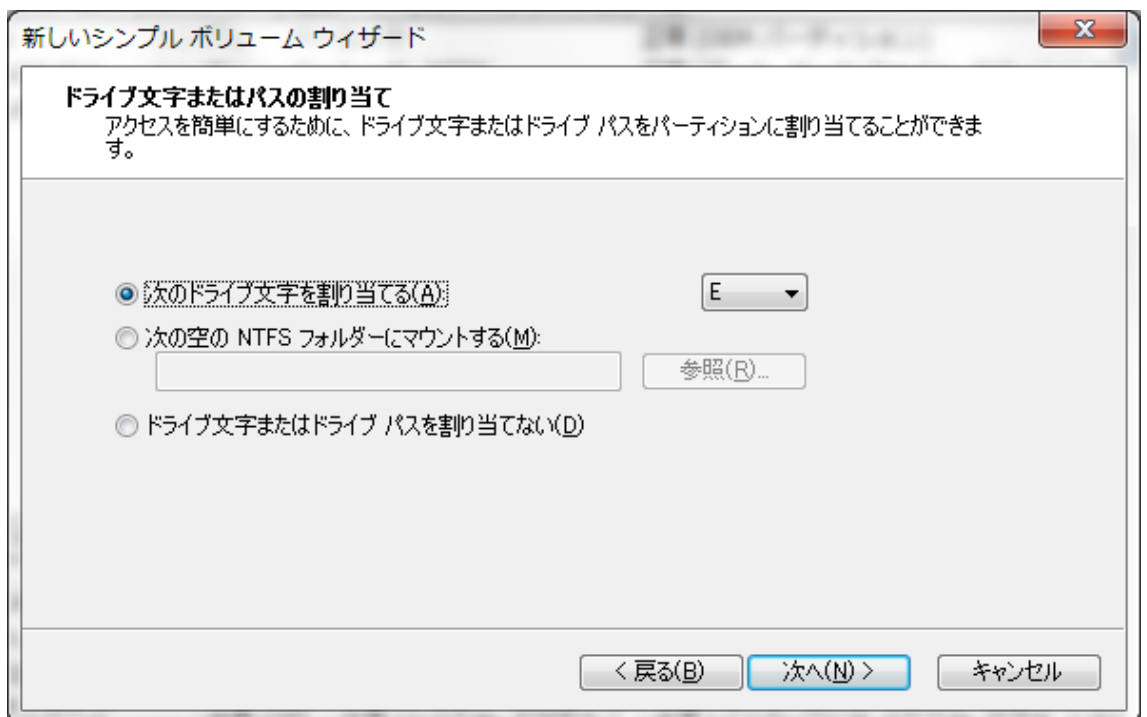
- ※ 「操作」 → 「タスク」 → 「新しいシンプルボリューム」か、マウス右クリック → 「新しいシンプルボリューム」で ウィザードが起動



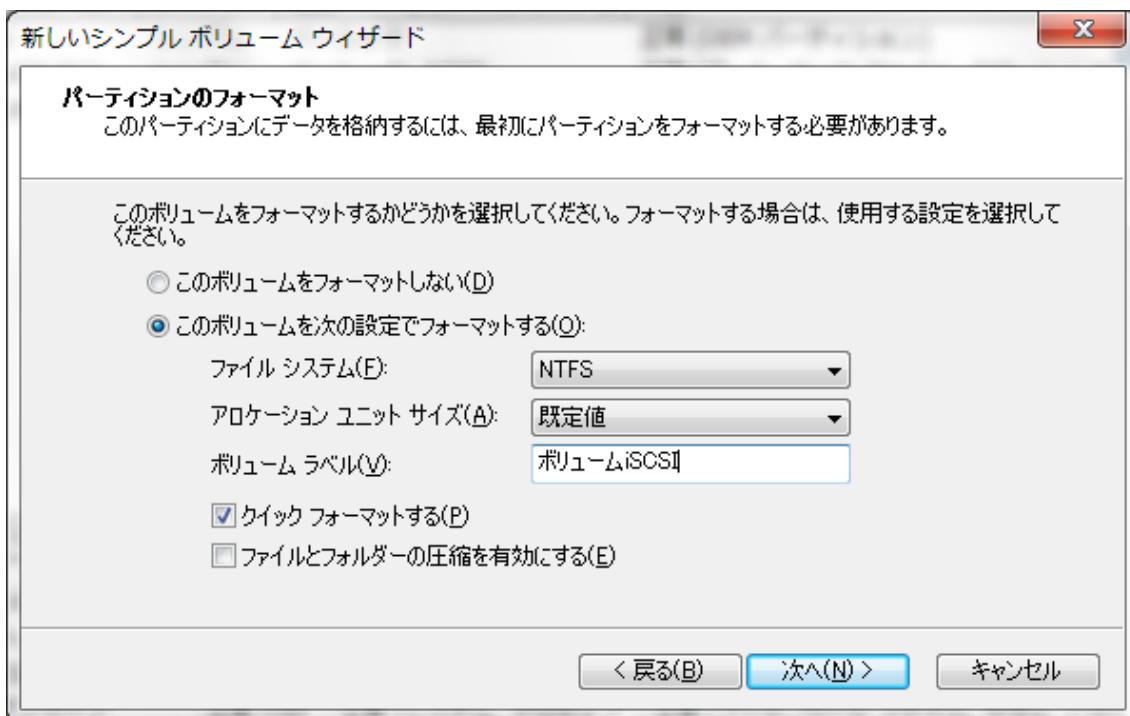
「次へ」ボタンを押します。



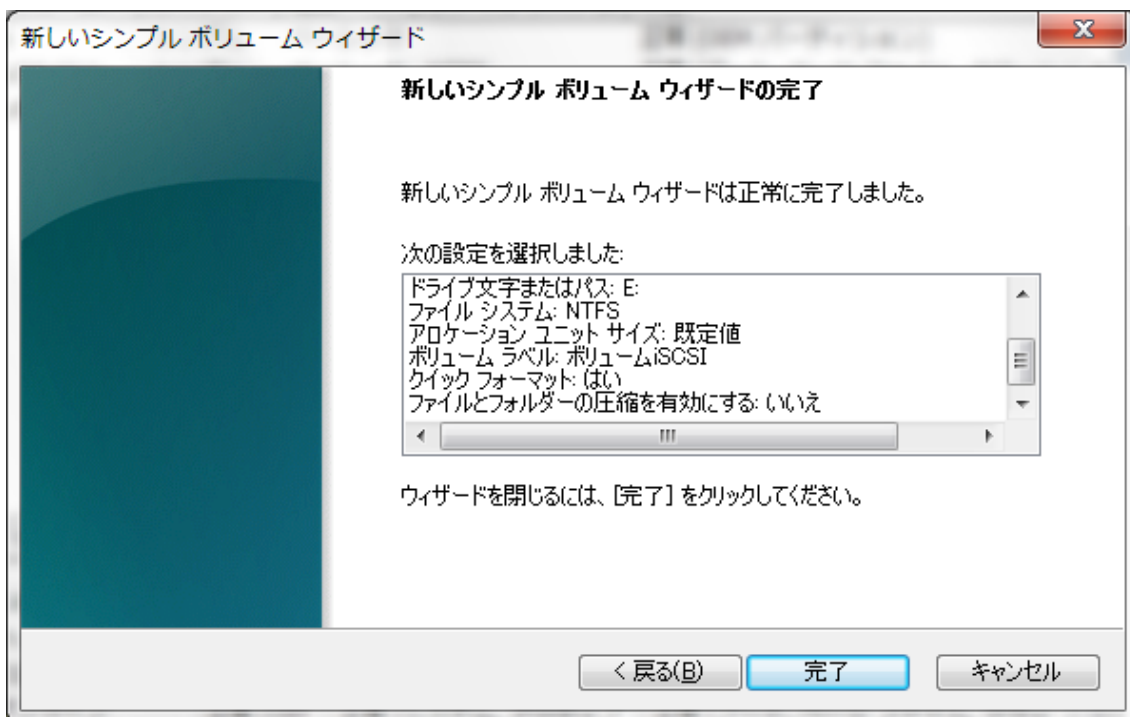
サイズを指定して、パーティション分けも可能ですが、例としてフルサイズを選択して「次へ」ボタンを押します。



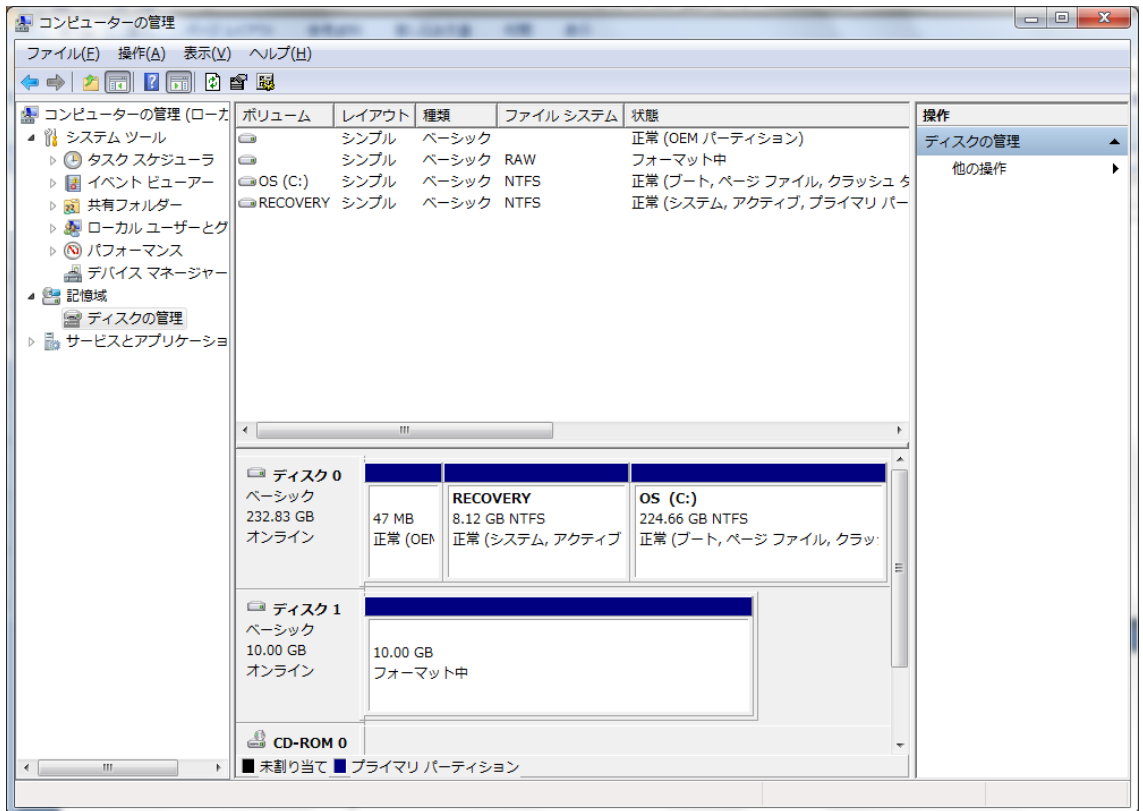
ドライブ追加する場合は、そのまま「次へ」ボタンを押します。



「ファイルシステム」「アロケーションユニットサイズ (ブロックサイズ)」「ボリューム名」を指定して「次へ」のボタンを押します。



問題がなければ「完了」を押します。



ウィザード終了して、「コンピュータの管理」の「ディスクの管理」画面で対象ディスクが「フォーマット中」と出ます。

終了したら、情報が表示される

