samba の設定

バグ？

基本的には GUI から設定して問題ない。
CentOS 5.2 の samba 設定 GUI ツールが死んでいるようだ。
ユーザーを登録しても、画面に表示されない。
コマンドからの設定は問題無し。

アクセスや動作が遅い

http://ftp3.samba.gr.jp/ml/article/samba-jp/msg14926.html
WinXP から共有にアクセスする際に、WebDAV によるアクセスを試しているらしい。
なので、ポート 80/tcp を閉じているとタイムアウト待ちになることがある。

80/tcp を開ける（apache 等のサービスは不要）と反応が早くなる。

ユーザーの管理（追加、表示、削除）

ユーザーの追加

```
pdbedit -a ユーザー名
```

ユーザーの表示

```
pdbedit -Lv
```

ユーザーの削除

```
pdbedit -x -u ユーザー名
```

グローバル設定

/etc/samba/smb.conf
http://d.hatena.ne.jp/korokorokoron/20100608/1275969242
http://linux2.g.hatena.ne.jp/pneumaster/
共有先にシンボリックリンクがある場合、以下の設定をしないとリンク先が開けない

```
[global]
wide links = yes
unix extensions  = no
```

■日付に関する設定

https://thinkit.co.jp/article/790/1?page=0%2C1

```
# タイムスタンプの精度を変更しない
dos filetime resoluntion = no
# ファイルの日付変更を書込権限のあるユーザに許可する
dos filetimes = yes
# ディレクトリの作成日をフェイクしない
#fake directory create times = yes
```

■実行権限に関する設定

http://www.dt8.jp/cgi-bin/adiary/adiary.cgi/0493

デフォルトの設定は、Windows のアーカイブ属性を所有者の実行権限とマッピングしている。
期待せずに実行権限が付いたり、外れたりするのでこの機能を無効にする。

```
map archive = no
```

共有の追加

/etc/samba/smb.conf

```
dos filetimes = Yes
dos filetime resolution = Yes
```

とすると、ファイルの更新日付を維持してコピーする

例

```
[user@private]
        path = /home/user/samba/private
        writeable = yes
        browseable = no
        create mask = 0600
        valid users = username
        directory mask = 0700


[user@share]
        path = /home/user/samba/share
        writeable = yes
        browseable = no
        create mask = 0660
        directory mask = 0770
```

ホームのパスを変更する例

```
[homes]
       path = /mnt/homes/%S
       valid users = %S
       略
```

ユーザ毎のディレクトリ（ホームっぽい）

```
[userdir]
      path = /mnt/userdir/%U
      valid users = %U
      略
```

ホームディレクトリを共有しない場合

ホームディレクトリを共有し無くない場合は
[homes] セクションをコメントアウトする。

```
;[homes]
  ;   comment = %U's Home Directory
  ;   browseable = no
  ;   writable = yes
  ;  create mask = 0644
  ;  directory mask = 0755
```

プリンタを共有に表示しない

http://fedorasrv.com/samba.shtml

[global] を編集する。

```
# If you want to automatically load your printer list rather
# than setting them up individually then you'll need this
;   load printers = yes  ←  行頭に ; を追加 ( プリンタ共有無効化 ) ※ Samba でプリンタを共有しない場合
  load printers = no  ←  追加 ( プリンタ共有無効化 ) ※ Samba でプリンタを共有しない場合
  disable spoolss = yes  ←  追加 ( プリンタ共有無効化 ) ※ Samba でプリンタを共有しない場合
```

シンボリックリンクのアクセスが拒否される場合

http://xoops.fens.net/modules/wiki/?Linux%2FMemo%2FSamba%2Fsymlink%20%E3%81%8C%E3%81%9F%E3%81%A9%E3%82%8C%E3%81%AA%E3%81%84

follow symlinks

共有エリア内へのシンボリックリンクを辿るかどうか

wide links

共有エリア外へのシンボリックリンクを辿るかどうか。 unix extensions = Yes の場合は動作しないので注意。

unix extensions

Samba が HP 社によって定義された CIFS の UNIX 拡張を利用するかどうかを制御する。 これらの拡張は、UNIX クライアントから UNIX サーバに対して接続する際に有用な CIFS を有効にし、 シンボリックリンクやハードリンクなどを利用可能にする。現在の Windows クライアントでは 有効になっていない。

もし、共有エリア外へのシンボリックリンクを辿る場合は

```
follow symlinks = Yes
wide links = Yes
#Samba-3.4.6/Samba-3.3.11 以降
unix extensions = No
```

とする必要がある。

アクセス権限のないファイルやディレクトリを隠す

samba の隠すには大きく２つある。

・見えないだけでアクセス可能。（パスを打ち込んだりすることでアクセス可能）
・見えないしアクセスも不能

■「.」で始まるファイルを隠す（アクセス可能）

```
hide dot files = yes
```

正確には隠しファイル属性を付与して表示する。

■読込権限のないファイル、ディレクトリを隠す（アクセス不能）

```
hide unreadable = yes
```

見えないし、パスを打ち込んでも無いことになっている。

```
hide unwriteable = yes
```

も設定可能らしいが、上手く動作しなかった。unreadable だけで十分な気もする。

パスワードなしでアクセスできる共有の設定

http://server-helper.doorblog.jp/archives/3152263.html
http://itpro.nikkeibp.co.jp/members/LIN/oss/20030924/1/

```
[global]
guest account = nobody
map to guest = Bad User
# 省略

[public]
guest ok = Yes
guest only = yes
# 省略
```

■ map to guest の値

設定値	意味
never（デフォルト）	ゲスト接続が許されず、登録されていないユーザーやパスワードが正しくないユーザーはアクセスできない。
bad user	登録されていないユーザーであっても、guest ok = Yes と定義された共有であればアクセスできる。
bad passwrod	登録されたユーザーがパスワードを間違えた場合、ゲストとして接続されることを意味する。したがって、guest ok = Yes と定義された共有であればアクセスできる。

■ guest ok = Yes / No

共有のゲストアクセスを許すかどうか指定する。この設定は、map to guest の指定がなされていないと有効にならない。

■ guest only = Yes

共有のアクセスをすべてゲスト権限にするかどうか指定する。この設定は、map to guest の指定がなされていないと有効にならない。

アクセス制御について

samba でのアクセス制御は大きく２つある
１．Linux のファイルパーミッションのみでの制御
２．samba の read list、write list、valid users、 invalid users による制御

１．Linux のファイルパーミッションのみでの制御

■メリット

・Linux のグループや SGID で制御するので、分かりやすくて設定しやすい。

■デメリット

複雑な制御ができない。

```
グループ A：読み書き可能
グループ B：読み込みのみ
```

など。

■例：各ユーザで共有するディレクトを作成する場合

http://blog.neet-shikakugets.com/article/202597386.html
http://jibun.atmarkit.co.jp/lskill01/rensai/lpicdrill09/lpicdrill02.html

```
[public]
    comment = Public Stuff
    path = /var/samba/public
    ;public = yes
    writable = yes
    printable = no
    create mask = 0664
    force create mode = 0664
    directory mask = 0775
    force directory mode = 0775
```

共有用のグループを作成

```
#groupadd smbPublic
```

共有用グループにユーザー追加

```
#gpasswd -a user1 smbPublic
```

か

```
#usermod -G smbPublic -a user1
```

どっちでもいいけど、gpasswd がおすすめかな。

/var/samba/public のグループ変更と sgid を設定

```
#chgrp smbPublic /var/samba/public
#chmod g+s /var/samba/public
```

create mask、 force create mode、 directory mask、 force directory mode については

```
作成されるファイルのパーミッション
(DOS から UNIX に設定される値 ) AND (create mask の値 ) OR (force create mode の値 )

作成されるディレクトリのパーミッション
(DOS から UNIX に設定される値 ) AND (directory mask の値 ) OR (force directory mode の値 )
```

２．samba の read list、write list、valid users、 invalid users による制御

■メリット

・複雑な制御ができる。

```
グループ A：読み書き可能
グループ B：読み込みのみ
```

などの複雑な制御ができる

・Linux のファイルパーミッションを併用できる。

■デメリット

・あまりやりすぎると分かり難くなる。
・smb.conf 内の共有設定毎にしか設定できない

■例：各ユーザで共有するディレクトを作成する場合

```
[public]
    comment = Public Stuff
    path = /var/samba/public
    printable = no
    create mask = 0664
    force create mode = 0664
    directory mask = 0775
    force directory mode = 0775
    write list = +smbPublic
    read list = +TestGroup
    valid users = +smbPublic +TestGroup
```

write list と read list の両方に含まれるユーザは、 write list が優先される（多分）
※ writable と read only を同時に書いた場合は、後に書いたほうが優先される（多分）

共有用のグループを作成

```
#groupadd smbPublic
```

共有用グループにユーザー追加

```
#gpasswd -a user1 smbPublic
```

■ valid users、invalid users、read list、write list に指定するグループ名

--	--
@ グループ名	NIS グループ、システムに設定してあるグループの順番で指定
& グループ名	NIS グループの指定
+ グループ名	システムに設定してあるグループの指定

認証について

http://blog.neet-shikakugets.com/article/202597386.html

```
Samba ユーザ -> Linux ユーザにマッピング -> アクセス
```

基本的には Samba ユーザと同じ名前の Linux ユーザが存在していないといけない。
[username map] や [username map script] パラメーターで対応を変更することが出来る。
Samba ユーザーは、あくまで認証情報や Windows ユーザーとしての情報を格納するためのもの。

/etc/samba/smb.conf の security パラメータに設定できるものは以下のもの。

値	認証の単位	説明
share	共有レベル	Windows NT または Samba にて共有単位で認証を行う
user	ユーザーレベル	Samba にて、ユーザー単位でのパスワード認証を行う
server	ユーザーレベル	Windows NT または Samba で認証を行う
domain	ユーザーレベル	Windows ドメインで認証を行う
ads	ユーザーレベル	Active Directory ドメインで認証を行う

Linux と Samba ユーザの一括管理

満足できる方法はないので省略。
http://kajuhome.com/tips/tips_06_001.shtml

NTLMv2 について

サーバ側の設定値と使用するプロトコル

パラメータの設定	認識可能な認証形式
encrypt passwords = No	平文パスワード
※ encrypt passwords = Yes / ※ lanman auth = Yes / ※ ntlm auth = Yes	NTLMv2/NTLM/LANMAN
※ encrypt passwords = Yes / lanman auth = No / ※ ntlm auth = Yes	NTLMv2/NTLM
※ encrypt passwords = Yes / ※ lanman auth = Yes / ntlm auth = No	NTLMv2/LANMAN
※ encrypt passwords = Yes / lanman auth = No / ntlm auth = No	NTLMv2

※「※」は Samba 3.0 系列のデフォルト値

クライアント側の設定値と使用するプロトコル

パラメータの設定	送信する認証形式
client ntlmv2 auth = Yes / - / - /	NTLMv2

※ client ntlmv2 auth = No / ※ client lanman auth = Yes / ※ client plaintext auth = Yes	NTLMv1/LANMAN/(平文 (※ 2))
※ client ntlmv2 auth = No / ※ client lanman auth = Yes / client plaintext auth = No	NTLMv1/LANMAN
※ client ntlmv2 auth = No / client lanman auth = No / -	NTLMv1

※「※」は Samba 3.0 系列のデフォルト値
※ 2: サーバが平文以外の認証方式に対応していない場合にのみ送信