

2008/9/10



2008 年 9 月 10 日開催

LPIC レベル 2 技術解説無料セミナー

日本電子専門学校

杉松 秀利

sugimatsu@mail.pcmarks.jp



Copyright(c) : Japan Electronics College All Rights Reserved.

LPIC レベル 2 技術解説無料セミナー

今回のセミナーでは、次の 4 つの項目をテーマにして解説します。

1. 出題範囲の把握
2. 受験対策
3. DNS サーバを構成する
4. NFS サーバを構成する

1. LPIC レベル 2 試験の出題範囲の把握

LPIC 201 試験と 202 試験の出題範囲は、次のとおりです。

(1) 201 試験の出題範囲

◆主題 201: Linux カーネル		◆主題 209: ファイルとサービスの共有	
2.201.1 カーネルの構成要素	1	2.209.1 Samba サーバーを構成する	5
2.201.2 カーネルのコンパイル	1	2.209.2 NFS サーバーを構成する	3
2.201.3 カーネルにパッチを当てる	2		
2.201.4 カーネルのカスタマイズ	1	◆主題 211: システムの保守	
		2.211.1 システムのログ	1
◆主題 202: システムの起動		2.211.2 ソフトウェアをパッケージ化する	1
2.202.1 システムの起動とブート手順を カスタマイズする	2	2.211.3 バックアップ操作	2
2.202.3 システムを回復する	3	◆主題 213: システムのカスタマイズと自動化	
		2.213.1 スクリプトを使って作業を自動化する	3
◆主題 203: ファイルシステム			
2.203.1 Linux ファイルシステムを操作する	3	◆主題 214 問題解決	
2.203.2 Linux ファイルシステムを保守する	4	2.214.2 回復ディスクを作成する	1
2.203.3 ファイルシステムの作成と オプションの設定	3	2.214.3 ブート段階を識別する	1
		2.214.4 ブートローダーの問題解決を行う	1
◆主題 204: ハードウェア		2.214.5 一般的な問題を解決する	1
2.204.1 RAID を構成する	2	2.214.6 システムリソースの問題を解決する	1
2.204.2 新しいハードウェアを追加する	3	2.214.8 環境設定の問題を解決する	1
2.204.3 ソフトウェアとカーネルを構成する	2		
2.204.4 PCMCIA デバイスを構成する	1		

(2) 202 試験の出題範囲

◆主題 205: ネットワーク設定		◆主題 210: ネットワーククライアントを管理する	
2.205.1 基本的なネットワーク構成	5	2.210.1 DHCP を構成する	2
2.205.2 上級のネットワーク構成と 問題解決	3	2.210.2 NIS を構成する	1
		2.210.3 LDAP を構成する	1
		2.210.4 PAM 認証	2
◆主題 206: メールとニュース		◆主題 212: システムのセキュリティ	
2.206.1 メーリングリストを構成する	1	2.212.2 ルータを構成する	2
2.206.2 メールサーバーを使用する	4	2.212.3 FTP サーバーのセキュリティ	2
2.206.3 メールトラフィックの管理	3	2.212.4 セキュアシェル (SSH)	2
2.206.4 ニュースサービス	1	2.212.5 TCPWrapper	1
◆主題 207: DNS		2.212.6 セキュリティ業務	3
2.207.1 DNS サーバーの基本的な構成	2	◆主題 214: ネットワークの問題解決	
2.207.2 DNS ゾーンを作成して保守する	3	2.214.7 ネットワークの問題を解決する	1
2.207.3 DNS サーバーのセキュリティ	3		
◆主題 208: Web サービス			
2.208.1 Web サーバーを実装する	2		
2.208.2 Web サーバーを保守する	2		
2.208.3 プロキシサーバーを実装する	2		

2. LPIC レベル 2 の受験対策

LPIC レベル 2 の受験対策としては、次のような方法を採用されると効果的です。

(1) レベル 2 の全般的な受験対策

- ① 試験範囲の内容を、どれだけ実機を使用して確認したかで、試験の合格/不合格は決定する。
- ② LPI-Japan の Web サイトの出題範囲の詳細ページに掲載されている「重要なファイル、用語、ユーティリティ」については、すべてチェックする。
- ③ コマンドについては、主要なオプションを整理する。
- ④ コマンドのオプションなどを調べる際には、英語版の man ページを参照することをお勧めします。
- ⑤ 実行可能なコマンドは、すべて実行してみる。
- ⑥ 設定ファイルについては、各エントリの定義の書式を整理し、設定可能なエントリを設定したうえで、実際にシステムに反映させてみる。
- ⑦ それぞれの設定ファイルの変更内容を、システムに反映させる操作方法も重要です。
- ⑧ 試験の出題範囲に含まれるサーバは、すべて構築してみる。
- ⑨ 以上のことを行うことによって、「主要な知識範囲」に指摘されている事項をクリアできるようにする。

(2) 201 試験の受験対策

- 試験 No.201 Linux 応用管理の受験対策は、次の 2 点に集約できます。
 - ① システムをカスタマイズする操作方法
 - ② システムをメンテナンス(管理/運用)する操作方法
- 実行可能なカスタマイズとメンテナンス操作は、すべて実行してみてください。
- その際、ご自分自身の実行操作のメモ(ノート)を作成することを、お勧めします。
- このときに作成したメモ(ノート)が、受験対策用の最適なテキストになります。
- レベル 2 試験の出題範囲のうちで、次のテーマについては、上記のような受験対策が適しています。
 - (1) 主題 201: Linux カーネル
 - (2) 主題 204: ハードウェア
 - (3) 主題 213: システムのカスタマイズと自動化
 - (4) 主題 214: 問題解決
 - (5) 主題 210: ネットワーククライアントを管理する
 - (6) 主題 214: ネットワークの問題解決

(3) 202 試験の受験対策

- 試験 No.202 Linux ネットワーク管理の受験対策としては、次の 2 つの内容が主要な基軸になります。
 - ① TCP/IP ネットワークに関する知識
 - ② 各種サーバの構築と管理/運用に関する知識

① TCP/IP ネットワークに関する知識に係る受験対策

- TCP/IP ネットワークを学習する機会の少ない方は、202 試験の個々のテーマの学習を始める前に、TCP/IP ネットワークに関する基礎的な内容について、学習することをお勧めします。
- 使用するテキストとしては、「マスタリング TCP/IP 入門編」(オーム社刊)が、現在刊行されているテキストの中では、最も優れたテキストであり、TCP/IP ネットワークに関する基礎的な知識について、体系的に理解できると思います。

② 各種サーバの構築と管理/運用に関する知識に係る受験対策

- 202 試験の出題範囲に含まれるサーバは、すべて構築してみることをお勧めします。
- 201 試験の受験対策と同様に、サーバ構築を行なう際には、ご自分自身の実行操作のメモ(ノート)を作成することをお勧めします。
- このときに作成したメモ(ノート)が、受験対策用の最適なテキストになります。
- サーバを構築する操作とは、ひとことで言ってしまうと、サーバの設定ファイルを編集する操作です。
- このため、各サーバの設定ファイルのエントリを定義する書式を整理しておくことが、重要になります。
- 出題範囲に含まれるサーバについては、サーバの設定ファイル、デーモン名、サービス制御スクリプト名およびコマンド名、およびそのサービスの起動/停止/再起動/状態の取得/設定ファイルの再読込などの動作オプションを、一括して整理しておくことをお勧めします。
- レベル 2 試験の出題範囲のうちで、次のテーマについては、上記のような受験対策が適しています。

- (1) 主題 202: システムの起動
- (2) 主題 203: ファイルシステム
- (3) 主題 209: ファイルとサービスの共有
- (4) 主題 211: システムの保守
- (5) 主題 205: ネットワーク設定
- (6) 主題 206: メールとニュース
- (7) 主題 207: DNS
- (8) 主題 208: Web サービス
- (9) 主題 212: システムのセキュリティ

3. DNS サーバを構成する

それでは、LPIC レベル 2 試験の出題範囲のうち、まずは 202 試験の「主題 207: DNS」から、

- (1) 2.207.1 DNS サーバの基本的な構成
- (2) 2.207.2 DNS ゾーンを作成して保守する
- (3) 2.207.3 DNS サーバのセキュリティ(chroot 環境での実行)
- (4) セカンダリ DNS サーバの設定方法

の概要について、学習することにしましょう。

(1) 2.207.1 DNS サーバの基本的な構成

BIND の基本的な構成要素は、次のとおりです。

◇BIND の構成要素

構成要素	ファイル名	摘要
基本設定ファイル	/etc/named.conf	BIND の基本設定ファイル
ゾーンファイル	/var/named/ディレクトリに格納される	個々のゾーンファイルの格納先
デーモン	named	named サービスのデーモン
サービス制御スクリプト	/etc/rc.d/init.d/named	BIND のサービス制御スクリプト
BIND のコマンド	/usr/sbin/named	BIND の実行コマンド
	/usr/sbin/ndc	BIND 8 で使用されるユーティリティコマンド
	/usr/sbin/rndc	BIND 9 で使用されるユーティリティコマンド

1) BIND における設定ファイルについて

- ① 基本設定ファイル: /etc/named.conf ファイル (/var/named/chroot/etc/named.conf ファイル)
 - DNS サーバのインデックスのような役割をするファイルです。
 - DNS サーバが管理するゾーンについては、zone ステートメントを使って、このファイル内に定義されていることが必要です。
 - また、このファイルには、DNS サーバの動作に関する各種のオプションが定義されます。
- ② ゾーンファイル
 - DNS サーバが管理するドメインの DNS 情報は、各ドメインに対応するゾーンファイルに定義されます。
 - ゾーンファイルには、次の 2 種類のゾーンファイルがあります。

- a. 正引きゾーンファイル … ホスト名・ドメイン名から IP アドレスへ名前解決するために使用する
- b. 逆引きゾーンファイル … IP アドレスからホスト名・ドメイン名へ名前解決するために使用する

2) BIND における基本設定ファイル/etc/named.conf の設定方法

◇BIND の設定ファイル/etc/named.conf ファイルの基本的な記述例

<pre>options { directory "/var/named"; }; zone "." IN { type hint; file "named.ca"; }; zone "example.co.jp" IN { type master; file "named.hosts"; }; zone "0.168.192.in-addr.arpa" IN { type master; file "named.hosts.rev"; }; zone "localhost" IN { type master; file "localhost.zone"; allow-update { none; }; }; zone "0.0.127.in-addr.arpa" IN { type master; file "named.local"; allow-update { none; }; };</pre>	<p>← options ステートメントでは、DNS サーバにの基本的な設定を記述</p> <p>← directory オプション には、ゾーンファイルが格納されるディレクトリを指定します。</p> <p>← zone ステートメントは、ゾーンの種類と、DNS 情報が格納されたファイル名などを定義します。</p> <p>“.”は、ルートドメイン(最上位)を示し、ここにはルートネームサーバに関する情報を記述します。</p> <p>hint は、キャッシュとして動作することを示します。</p> <p>named.ca ファイルには、ルートネームサーバに関する DNS 情報が記述されています。</p> <p>← example.co.jp の正引きゾーンに関する定義であることを示す。</p> <p>← 注</p> <p>← ゾーンファイル名が named.hosts であることを指定しています。</p> <p>← 192.168.0.0 の逆引きゾーンに関する定義であることを示しています。</p> <p>← 注</p> <p>← ゾーンファイル名を指定しています。</p> <p>逆引きゾーンの指定は、IP アドレスを逆に書き、最後に.in-addr.arpa と書きます。</p> <p>← localhost の正引きゾーンに関する定義であることを示しています。</p> <p>← 注</p> <p>← ゾーンファイル名を指定しています。</p> <p>← ダイナミック(Dynamic)DNS の使用を禁止することを指定しています。</p> <p>BIND 8 から、ダイナミック(Dynamic)DNS 機能が追加されています。</p> <p>← ローカル・ループバックアドレス(127.0.0.1)の逆引きゾーンに関する定義であることを示しています。</p> <p>← ゾーンファイル名を指定しています。</p> <p>← ダイナミック(Dynamic)DNS の使用を禁止することを指定しています。</p>
--	--

```
include "/etc/rndc.key";
```

← named.conf を参照するときに、一緒に読み込むファイル名を指定しています。

注: 「type master;」について

ゾーンの種類として master を指定した場合には、DNS サーバが対象となるゾーンの DNS 情報のマスタファイルを保持しており、そのゾーンについて信頼できる応答を提供できます。

a. version オプション

実際とは異なるバージョン番号を返すための BIND の設定として、最もシンプルな方法は、/etc/named.conf ファイル内の options ステートメントで、version オプションに任意の値を指定することです。

```
options {
    directory "/var/named";
    (中略)

    version " my version ";

    (後略)
};
```

b. forwarders オプション

- forwarders オプションには、DNS サーバが名前解決できない場合に、再帰的問い合わせを行う特定の外部の DNS サーバの IP アドレスを指定します。
- forwarders オプションも、/etc/named.conf ファイル内の options ステートメント内に指定します。

◇forwarders オプションの記述例

```
forwarders { 211.129.14.138; };
```

c. その他の主要なオプション

◇/etc/named.conf ファイルに定義するセキュリティオプション

オプション	説明
allow-transfer	ゾーン転送を許可するホストを指定する
allow-query	問い合わせを受け付けるホストを指定する
allow-recursion	再帰的な問い合わせを受け付けるホストを指定する
blackhole	問い合わせを受け付けないホストを指定する

◇記述例

```
allow-transfer { 192.168.0.10; };
allow-query { 192.168.0.0/24; };
allow-recursion { 192.168.0.0/24; };
blackhole { 192.168.0.50; };
```

3) 変更した設定ファイルやゾーンファイルを再読みする

変更した基本設定ファイルやゾーンファイルを再読みする場合には、次の2つの操作方法があります。

a. /usr/sbin/rndc コマンドを使用する方法

◇/usr/sbin/rndc コマンドを使用する場合

```
# rndc reload | restart
```

注:

- /usr/sbin/rndc コマンドを実行すると、実際には/usr/sbin/named コマンドが実行されます。
- reload オプションを指定した場合には、サーバを再起動することなく、設定ファイルのみを再読みします。
- restart オプションを指定した場合には、サーバを再起動して、設定ファイルを再読みします。

b. /etc/rc.d/init.d/named スクリプトを使用する方法

◇/etc/rc.d/init.d/named スクリプトを使用する場合

```
# /etc/rc.d/init.d/named reload | restart
```

注: /etc/rc.d/init.d/named スクリプト内では、実際には/usr/sbin/named コマンドが実行されます。

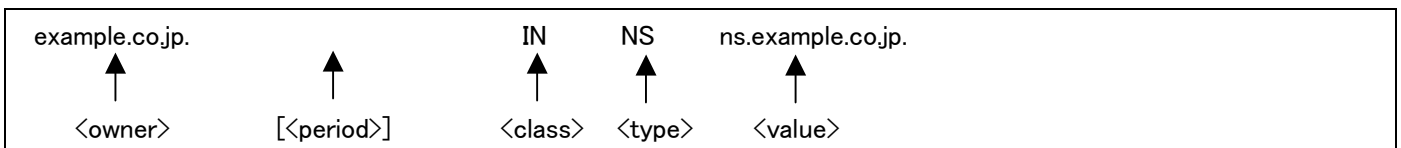
(2) 2.207.2DNS ゾーンを作成して保守する

このテーマでは、正引きならびに逆引きのゾーンファイルの作成方法、および既存のゾーンファイルにリソースレコードを追加する操作方法に関する知識などが問われます。

1) リソースレコードの書式

ゾーンファイルの内容は、リソースレコードを使用した定義の集合体です。

◇リソースレコードの書式



◇各フィールドの説明

項目	指定内容
<owner>	ドメイン名やホスト名などを指定します。
<period>	有効期限(TTL 値)を指定するフィールドですが、省略できます。
<class>	通常の場合、インターネットを表すINを指定します。
<type>	リソースレコードの種類を指定します。
<value>	リソースレコードの種類に応じた値を指定します。

2) 主要なリソースレコードの種類

◇ゾーンファイルで使用する主なリソースレコード

レコード名	意味	説明
SOA	Start Of Authority	ゾーンデータの保守/運用に関する情報を定義する
NS	Name Server	ゾーンの DNS 情報を管理するネームサーバ名を定義する
MX	Mail eXchanger	ゾーンのメール処理を行うホスト名を定義する
A	Address	ホスト名から IPv4 の IP アドレスへのマッピングを定義する
AAAA	Quad A	ホスト名から IPv6 の IP アドレスへのマッピングを定義する
CNAME	Canonical Name	ホストの別名(エイリアス)を定義する
PTR	Domain Name Pointer	IP アドレスからホスト名へのマッピングを定義する

3) 正引きゾーンファイルの設定方法

◇正引きゾーンファイルの記述例

```

$TTL 86400
@           IN  SOA  ns.example.co.jp.  postmaster.example.co.jp. (
                2008091001      ;Serial
                28800           ;Refresh
                14400           ;Retry
                3600000         ;Expire
                86400 )         ;Minimum TTL

; Authoritative Name Servers
                IN  NS      ns.example.co.jp.
                IN  NS      ns.sub-example.co.jp.

; Mail eXchanger
                IN  A       192.168.0.30
                IN  MX      10 mail.example.co.jp.

; Hosts
ns.example.co.jp.  IN  A       192.168.0.20
www.example.co.jp. IN  CNAME   ns.example.co.jp.
mail.example.co.jp. IN  A       192.168.0.30
ftp.example.co.jp. IN  A       192.168.0.100

; Localhost
localhost.example.co.jp. IN  A       127.0.0.1

```

◆SOAレコードの記述内容

			① ネームサーバ名	② 管理者のメールアドレス
@	IN SOA	ns.example.co.jp.	postmaster.example.co.jp.	(
	2008073001		; ③ シリアル番号 (Serial)	
	28800		; ④ リフレッシュ間隔 (Refresh)	
	14400		; ⑤ リトライ間隔 (Retry)	
	3600000		; ⑥ レコード有効期間 (Expire)	
	86400)		; ⑦ ネガティブキャッシュ有効期間 (Minimum)	

ドメイン定義を宣言する SOA レコードには、具体的には、次のような情報を記述します。

- ① ドメインの DNS サーバ名
- ② ドメイン管理者のメールアドレス
- ③ シリアル番号 (Serial)

ゾーン転送時に、情報が更新されているかどうかを判断するために使用されます。数値が大きくなっていれば、更新済みであることを意味します。指定する番号は任意ですが、管理しやすいように、通常の場合には

年月日 + 連番

という書式で指定します。

- ④ リフレッシュ間隔 (更新間隔) (Refresh)

セカンダリ DNS サーバが、プライマリ DNS サーバの情報に変更がないかを確認するために、シリアル番号をチェックする間隔を秒単位で指定します。

- ⑤ リトライ間隔 (転送再試行時間) (Retry)

ゾーン転送に失敗した場合に、再試行を実行するまでの猶予時間を秒単位で指定します。

- ⑥ レコード有効時間 (Expire)

セカンダリ DNS サーバが、プライマリ DNS サーバと通信できない場合に、セカンダリ DNS サーバが保持するゾーンデータの有効時間を秒単位で指定します。

- ⑦ ネガティブキャッシュ (Minimum)

DNS 問い合わせを行った結果、ホストが存在しないなどの理由によって名前解決に失敗した場合に、名前解決ができなかったという事実とそのレコード情報を、キャッシュに保存しておく時間を秒単位で指定します。

注: 上記の正引きゾーンファイルの記述例の 1 行目の「\$TTL 86400」は、デフォルトの TTL 値を指定しています。

上記の正引きゾーンファイルの記述例におけるリソースレコードの定義は、次のようになっています。

; Authoritative Name Servers			
	IN NS	ns.example.co.jp.	← NS レコード ①
	IN NS	ns.sub-example.co.jp.	← NS レコード ②
; Mail eXchanger			
	IN A	192.168.0.30	
	IN MX 10	mail.example.co.jp.	← MX レコード

; Hosts				
ns.example.co.jp.	IN	A	192.168.0.20	← A レコード ①
www.example.co.jp.	IN	CNAME	ns.example.co.jp.	← CNAME レコード
mail.example.co.jp.	IN	A	192.168.0.30	← A レコード ②
ftp.example.co.jp.	IN	A	192.168.0.100	← A レコード ③
; Localhost				
localhost.example.co.jp.	IN	A	127.0.0.1	← A レコード ④

4) 逆引きゾーンファイルの設定方法

◇逆引きゾーンファイルの記述例

```

$TTL 86400
@           IN  SOA  ns.example.co.jp.  postmaster.example.co.jp. (
                2008091001      ;Serial
                28800            ;Refresh
                14400            ;Retry
                3600000          ;Expire
                86400 )         ;Minimum TTL

; Authoritative Name Servers
                IN  NS      ns.example.co.jp.
                IN  NS      ns.sub-example.co.jp.

;
20           IN  PTR      ns.example.co.jp.
30           IN  PTR      mail.example.co.jp.
100          IN  PTR      ftp.example.co.jp.

```

上記の記述例のように、逆引きゾーンファイルは、通常の場合には、SOA レコード、NS レコード、PTR レコードの定義によって構成されています。

<type>	意味	説明
SOA	Start Of Authority	ゾーンデータベースの保守/運用に関する情報を定義するレコード
NS	Name Server	ゾーンの DNS 情報を管理するネームサーバ名を定義する
PTR	Domain Name Pointer	IP アドレスからホスト名へのマッピングを定義するレコード

5) ゾーンファイルの保守

- ゾーンファイルの設定内容を変更した場合には、SOA レコードのシリアル番号を変更する必要があります。
- 変更するシリアル番号には、これまでの番号よりも、大きい値を設定する必要があります。

◎ サブドメインへの権限委譲の設定

サブドメインへの権限委譲を行う場合には、親ドメインのゾーンファイルの中に、サブドメインのネームサーバ名を指定するために、NSレコードとAレコードを定義しなければなりません。

◇ 権限委譲を定義するリソースレコードの書式

サブドメイン名.	IN NS	サブドメインの DNS サーバ名.	
....			
サブドメインの DNS サーバ名.	IN A	IP アドレス	← グルーレコード

注:

権限委譲を行なうために、親ドメインのゾーンファイル内に記述したサブドメインのネームサーバ名の IP アドレスを定義する A レコードの記述のことを、「グルーレコード」と言います。

◎ リゾルバの設定

- ① 各ホストから、名前解決を行うために問い合わせを行う DNS サーバは、リゾルバ(name resolver)という機能によって設定します。
- ② リゾルバは、`/etc/resolv.conf` ファイルを使って設定します。

リゾルバを設定する場合には、`/etc/resolv.conf` ファイルの中に、問い合わせ先の DNS サーバの IP アドレスを、次のような書式で記述します。

◇ resolv.conf ファイルの記述例①

nameserver 10.1.1.1	← プライマリ DNS サーバの IP アドレスを指定しています。
nameserver 10.1.1.2	← セカンダリ DNS サーバの IP アドレスを指定しています。

◇ resolv.conf ファイルの記述例②

search cljtec.ac.jp	← 検索するドメイン名を指定しています。
nameserver 10.40.192.200	← 1 番目の DNS サーバの IP アドレスを指定しています。
nameserver 10.24.96.200	← 2 番目の DNS サーバの IP アドレスを指定しています。
nameserver 10.1.1.1	← 3 番目の DNS サーバの IP アドレスを指定しています。

注: 3 台までの DNS サーバの IP アドレスが指定でき、記述された順番で問い合わせが行なわれます。

(3) 2.207.3 DNS サーバのセキュリティ(chroot 環境での実行)

- chroot(チェンジルート)とは、実行中の BIND から参照できる最上位のパスを、下位のディレクトリに変更した環境のことです。この環境では、BIND を経由して外部から攻撃を受けた場合でも、主要なファイルが格納されている上位のディレクトリが見えないため、被害を限定的に止めることができます。
- ただし、悪意の第三者がプライマリ DNS サーバになりすますことを、直接的に排除するための対策法ではないので、偽の DNS 情報を転送する危険性を排除する方法としては、有効ではありません。

◇chroot のファイルシステム上の構成

```

/
|
|-- etc/ named.conf ← /var/named/chroot/etc/named.conf ファイルのシンボリックリンク
|
|-- var/
|   |-- named/ ゾーンファイル ← chroot/etc/named/ディレクトリに格納されているファイルのシンボリックリンク
|   |   |-- chroot/
|   |   |   |-- etc/ named.conf ← /etc/named.conf ファイルのファイル実体
|   |   |   |-- var/
|   |   |   |-- named/ ゾーンファイル ← /var/named/ディレクトリのゾーンファイルのファイル実体

```

(4) セカンダリの DNS サーバの設定方法

セカンダリ DNS サーバを設定する場合には、次の 2 つの設定操作を行ないます。

- ① /etc/named.conf ファイルの設定 (/var/named/chroot/etc/named.conf ファイルの設定)
- ② ゾーンファイルが保存される/var/named/chroot/var/named/ディレクトリの所有者の変更

① /etc/named.conf ファイルの設定

/etc/named.conf ファイルの設定では、DNS サーバがセカンダリ DNS サーバとして動作するために、

- 1) 正引きゾーンの定義
- 2) 逆引きゾーンの定義

を行ないます。

1) 正引きゾーンの定義

セカンダリ DNS サーバの正引きゾーンの定義は、次のように記述します。

◇正引きゾーンを定義するための zone ステートメントの記述例

zone "example.co.jp" IN {	← プライマリ DNS サーバに定義した正引きゾーン名
type slave;	← ゾーンの種類を slave に設定することに注意!
masters {	
192.168.0.20;	← プライマリ DNS サーバの IP アドレスを指定する
};	
file "named.hosts.bak";	← プライマリ DNS サーバのゾーンファイル名 + .bak
allow-update { none; };	
};	

注:

- ① セカンダリ DNS サーバで管理するゾーンファイル名については、プライマリ DNS サーバの場合と同様に、特別な命名規則はありません。
- ② ただし、プライマリ DNS サーバで管理するゾーンファイルとの関連性が分かるようなファイル名にしておく方が、効率的にメンテナンスが行なえます。

2) 逆引きゾーンファイルの定義

セカンダリ DNS サーバの逆引きゾーンの定義は、次のように記述します。

◇逆引きゾーンを定義するための zone ステートメントの記述例

```
zone "0.168.192.in-addr.arpa" IN {
    type slave;
    masters {
        192.168.0.20;
    };
    file "named.hosts.rev.bak";
    allow-update { none; };
};
```

← プライマリ DNS サーバに定義した逆引きゾーン名
 ← ゾーンのタイプを slave に設定することに注意！
 ← プライマリ DNS サーバの IP アドレスを指定する
 ← プライマリ DNS サーバのゾーンファイル名 + .bak

② ゾーンファイルが保存されるディレクトリの所有者の変更

一般的には、サービス・デーモン named は、次のステートメントを使って実行されます。

◇named を実行するステートメント

```
# /usr/sbin/named -u named -t /var/named/chroot
```

注： サービス制御スクリプトを使って、次のステートメントで実行した場合でも、実際には上記のステートメントが実行されます。

◇サービス制御スクリプトにより named を実行するステートメント

```
# /etc/rc.d/init.d/named start
```

また、chroot 環境でゾーンファイルが格納されるディレクトリの所有者は、デフォルトでは次のように設定されています。

```
# ls -l /var/named/chroot/var/
drwxr-x--- 4 root named 4096 Jul 12 00:16 named ← このディレクトリの所有者
      ↓
drwxr-x--- 4 named named 4096 Jul 12 00:16 named ← named:named に変更する
drwxrwx--- 3 root named 4096 Jul 12 00:16 run
drwxrwx--- 2 named named 4096 Mar 14 2003 tmp
```

デフォルトの所有者の設定のままでは、プライマリ DNS サーバとセカンダリ DNS サーバ間でゾーン転送が行なわれた場合に、プロセスの所有者が named であるため、プライマリ DNS サーバから転送されるゾーンファイルを、当該ディレクトリに保存することができないために、ゾーン転送は失敗してしまいます。

そこで、転送されるゾーンファイルを保存する /var/named/chroot/var/named/ ディレクトリの所有者を、次のステートメントを実行することによって、named に変更します。

◇ゾーンファイルが保存されるディレクトリの所有者を変更するステートメント

```
# chwon named.named /var/named/chroot/var/named/
```

以上で、セカンダリ DNS サーバの設定は終了です。

通常の DNS サーバの設定では、上記の設定操作を行なったのちに、プライマリ DNS サーバの `named` デーモンを再起動(または設定ファイルを再読み込み)し、その後にセカンダリ DNS サーバの `named` デーモンを起動して、ゾーン転送が行なわれるかどうかを確認します。

③ ゾーン転送に係るトラブルシューティングのヒント

ゾーン転送が正常に行なわれない場合には、次の 2 点について確認します。

- ① TCP の 53 番ポートが、経路の途中でフィルタリングされていないかを確認する。
 - ゾーン転送の通信は、DNS 問い合わせの応答の場合と異なり、UDP ではなく TCP を使って行われます。
 - プライマリとセカンダリの DNS サーバの間に、ファイアウォールが設置されていたり、`iptables` などによってパケット・フィルタが設定されている場合には、該当するポートを開放する操作を行うことが必要になります。
- ② プライマリ DNS サーバにおいて、ゾーン転送の制限を設定していないかを確認する。

4. NFS サーバを構成する

続いて、201 試験の「主題 209:ファイルとサービスの共有」から、「2.209.2 NFS サーバを構成する」の概要について、学習することにしましょう。

(1) NFS の設定ファイル

NFS サーバと NFS クライアントを設定するためには、次の 4 つの設定ファイルを編集します。

- ① /etc/exports NFS サーバを設定するファイル
- ② /etc/hosts 名前解決を行うための IP アドレス、ホスト名を定義するファイル
- ③ /etc/hosts.deny サーバへのアクセス拒否を設定するファイル
- ④ /etc/hosts.allow サーバへのアクセス許可を設定するファイル

(2) NFS のツールとユーティリティ

NFS を操作するために、次のようなツールとユーティリティが提供されています。

① exportfs コマンド

現在 NFS でエクスポートしているファイルシステムのテーブルを管理するために使用するコマンドです。

◇exportfs コマンドの主要なオプション

オプション		指定内容
-a	all	すべてのディレクトリを、エクスポート(またはアンエクスポート)する。
-r	reexport	すべてのディレクトリを再エクスポートする。
-u	unexport	アンエクスポートする。
-v	verbose	詳細を表示する

② showmount コマンド

NFS サーバが正しく起動されたかどうかを確認するために使用するコマンドです。

◇showmount コマンドの書式

showmount -e NFS サーバのホスト名(または IP アドレス)
--

③ nfsstat コマンド

NFS クライアントとサーバの動作に関して保存されている統計を表示するために使用するコマンドです。

◇nfsstat コマンドの主要なオプション

オプション	指定内容
-s	サーバ側の統計のみを表示する。デフォルトではサーバとクライアント両者の統計を表示する。
-c	クライアント側の統計のみを表示する。
-n	NFS の統計のみを表示する。デフォルトでは、NFS と RPC 両者の情報を表示する。
-r	RPC の統計のみを表示する。
-o facility	引数 facility に指定した統計のみを表示する。以下のうちのひとつを指定できる。 nfs ... RPC コールを除く、NFS プロトコルの情報 rpc ... 一般的な RPC 情報 net ... ネットワーク層の統計。例えば受信パケットの数、TCP 接続回数など。

fh	...	サーバのファイルハンドルキャッシュの利用情報。ルックアップの回数、ヒットとミスの回数を含む。
rc	...	サーバのリクエスト返信用キャッシュの利用情報。ルックアップの回数、ヒットとミスの回数を含む。

◎ 構成前の確認事項

- NFS サービスを利用するためには、RPC サービスプログラムに動的に TCP/UDP ポート番号を割り当てるためのプロトコルである portmap (ポートマップ) が、NFS サーバと NFS クライアントの両方で、起動していなければなりません。
- このため、NFS サーバに設定するホストと、NFS クライアントとして使用するホストの両方で、現在、portmap が起動しているかどうかを確認してください。
- 次のステートメントを実行して、確認操作を行ってください。

◇portmap の起動を確認するステートメントの実行例

```
# ps aux | grep portmap
rpc      493  0.0  0.3 1528  600? S    15:28   0:00 portmap
```

- もし、portmap が起動していない場合には、次のステートメントを実行して、portmap サービスを起動します。

◇portmap サービスを起動するステートメント

```
# /etc/rc.d/init.d/protmap start
```

(3) 設定ファイル/etc/exports の設定方法

設定ファイル/etc/exports の設定は、次の操作手順で行ないます。

① エクスポート・ディレクトリの作成

- NFS サーバ側のホストに、たとえば public という名前の新しいディレクトリを作成します。
- このディレクトリを使って、NFS クライアント側のホストとの間で、ディレクトリ共有を行います。
- NFS では、他のホストと共有するために公開するディレクトリのことを、エクスポート・ディレクトリと言います。

◇エクスポートするディレクトリを作成する操作例

```
# cd /
# mkdir /public
```

◇ 確認用のテキストファイルの作成

作成したエクスポート・ディレクトリに、vi などのエディタを使ってテキストファイルを 1 個作成してください。

② /etc/exports ファイルの編集

- 次に、NFS クライアントへ公開するエクスポート・ディレクトリに関する情報を設定します。
- 作成または編集する設定ファイルは、/etc/exports ファイルです。
- /etc/exports ファイルに記述する書式は、次のとおりです。

◇書式

エクスポートするディレクトリ名 アクセスを許可するホスト名 | IP アドレス(オプション)

○主要なオプション(ディレクトリ共有の権限の指定)

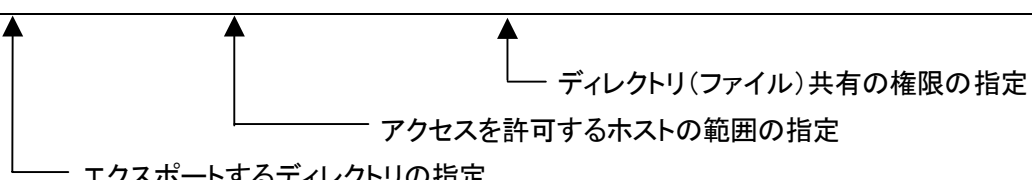
オプション	機能
ro	ディレクトリを、読み取り専用でエクスポートする。(デフォルトの設定値)
rw	ディレクトリを、読み取り、書き込み許可で、エクスポートする。
root_squash	root 権限によるアクセスを、nobody というユーザからのアクセスとして扱う。(デフォルトの設定値)
no_root_squash	root 権限でのアクセスを許可する。
noaccess	指定したディレクトリ以下のアクセスを禁止する。

注: ro は read only、rw は read write です。

◇vi などを使って/etc/exports ファイルを開き、次のように記述します。

①特定のネットワーク上のホストとの間で、ディレクトリ共有を行う場合の記述例

```
/public            192.168.0.0/255.255.255.0(rw)
```



②特定のホストとの間で、ディレクトリ共有を行う場合の記述例

```
/public    192.168.0.51(rw)
```

③複数のホストとの間で、ディレクトリ共有を行う場合の記述例 (スペースで区切って指定する)

```
/public    192.168.0.51(rw) 192.168.0.52(ro)
```

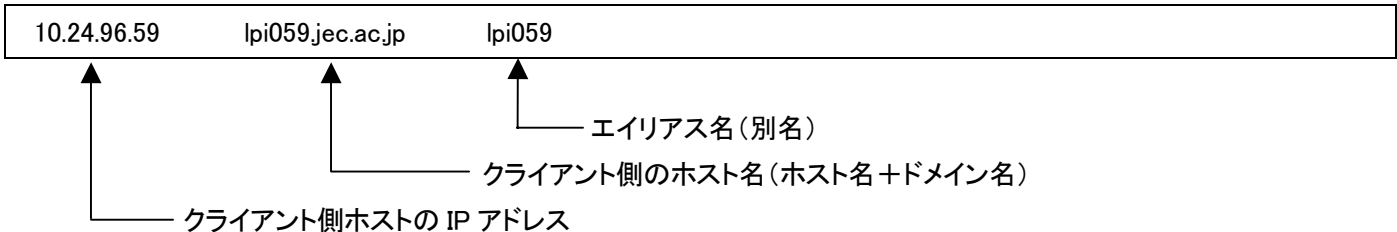
④すべてのホストとの間で、ディレクトリ共有を行う場合の記述例

```
/public    *(ro)
```

③ /etc/hosts ファイルの編集

ローカルホスト(自ホスト)内で、NFS サーバにアクセスするホスト(NFS クライアント)について、名前解決が出来るようにするために、IP アドレス、ホスト名、エイリアス名を、/etc/hosts ファイルに定義します。

◇vi などを使って/etc/hosts ファイルを開き、次のような1行を追加する。



(4) TCP ラッパーの設定方法

- 続いて、TCP ラッパーによるアクセス制御を設定します。
- TCP ラッパーの設定には、次の 2 つの設定ファイルを使用します。
 - ① /etc/hosts.deny ... アクセス拒否の設定を行なうファイルです。
 - ② /etc/hosts.allow ... アクセス許可の設定を行なうファイルです。

① /etc/hosts.deny ファイルの編集

NFS サーバへのアクセス拒否を設定するために、/etc/hosts.deny ファイルを編集します。

◇/etc/hosts.deny ファイルと/etc/hosts.allow ファイルの書式

```
サービス名: ホストの IP アドレス | ネットワークアドレス/サブネットマスク | ALL , . . .
```

◇portmap サービスについて、すべてのホストからのアクセスを拒否する指定

```
portmap : ALL
```

◇すべてのサービスについて、すべてのホストからのアクセスを拒否する指定

```
ALL : ALL
```

【注意事項】

/etc/hosts.deny の記述を最後まで確認して、上記の 1 行が記述されていない場合にのみ、追加してください。

② /etc/hosts.allow ファイルの編集

NFS サーバへのアクセス許可を設定するために、/etc/hosts.allow ファイルを編集します。

◇portmap サービスについて、特定のネットワーク上のホストからのアクセスのみを許可する指定

```
portmap : 10.24.96.0/255.255.255.0
```

(5) NFS サービスの起動

以上の設定が終了したら、NFS サービスを起動します。

◇NFS サービスを起動するステートメント

```
# /etc/rc.d/init.d/nfs start
Starting NFS services:           [ OK ]
Starting NFS quotas:           [ OK ]
Starting NFS daemon:           [ OK ]
Starting NFS mounted:         [ OK ]
```

◇NFS サービスの起動後に、現在の状態を確認した実行例

```
# /etc/rc.d/init.d/nfs status
rpc.mountd (pid 6415) is running . . .
nfsd (pid 6407 6406 6405 6404 6403 6402 6401 6400) is running . . .
rpc.rquotad (pid 6395) is running . . .
```

◇NFS サービスを再起動するステートメント

```
# exportfs -a                ← エクスポートの設定を初期化する(アンエクスポートする)
# /etc/rc.d/init.d/nfs restart ← nfs サービスを再起動する
```

(6) クライアントからのマウント操作

NFS クライアントから NFS サーバへマウントを行なう場合には、次のような操作手順で行ないます。

① 起動内容の確認

NFS サーバが、正しく起動されたかどうかを確認するために、次の操作を実行します。使用するのは、showmount コマンドです。

◇showmount コマンドの書式

```
# showmount -e NFS サーバのホスト名(または IP アドレス)
```

◇NFS サーバが正常に起動しているかどうかを確認する操作例

```
# showmount -e localhost
Export list for localhost:
/public    192.168.0..0/255.255.255.0
```

② エクスポート・テーブルの参照

- エクスポート・テーブルを操作する場合には、/usr/sbin/exportfs コマンドを使用します。
- 現在のエクスポート・テーブルの内容は、exportfs コマンドに -v オプションを指定して実行すると、詳細に表示できます。

◇エクスポート・テーブルを参照する操作の実行例

```
# exportfs -v
/public 10.24.96.0/255.255.255.0 (rw,wdelay,root_squash)
```

③ /etc/hosts ファイルの編集

NFS サーバのホストについて、ローカルホスト内で名前解決ができるようにするために、/etc/hosts ファイルを編集します。

◇/etc/hosts ファイルの記述例

127.0.0.1	localhost.localdomain	localhost	← ループバックアドレスの記述
10.24.98.51	lpi051.jec.ac.jp	lpi051	← 自ホストに関する記述
10.24.96.55	lpi055.jec.ac.jp	lpi055	← NFS サーバに関する記述

④ NFS クライアントからの NFS サーバへのマウント操作

- それでは、NFS クライアントから NFS サーバへマウントしてみましょう。
- mount コマンドを使って、NFS サーバのエクスポート・ディレクトリへ NFS マウントします。

◇NFS マウントを行なうための mount コマンドの書式

```
# mount -t nfs ホスト名または IP アドレス:/エクスポート・ディレクトリ名 マウントポイント
```

◇NFS マウントを行なうための mount コマンドの実行例①

```
# mount -t nfs lpi055:/public /media
```

◇NFS マウントを行なうための mount コマンドの実行例②

```
# mount -t nfs 10.24.96.55:/public /media
```

⑤ NFS マウントの終了操作

NFS マウントを終了する場合には、umount コマンドを使用します。

◇NFS マウントを終了するための umount コマンドの書式

```
# umount マウントポイント
```

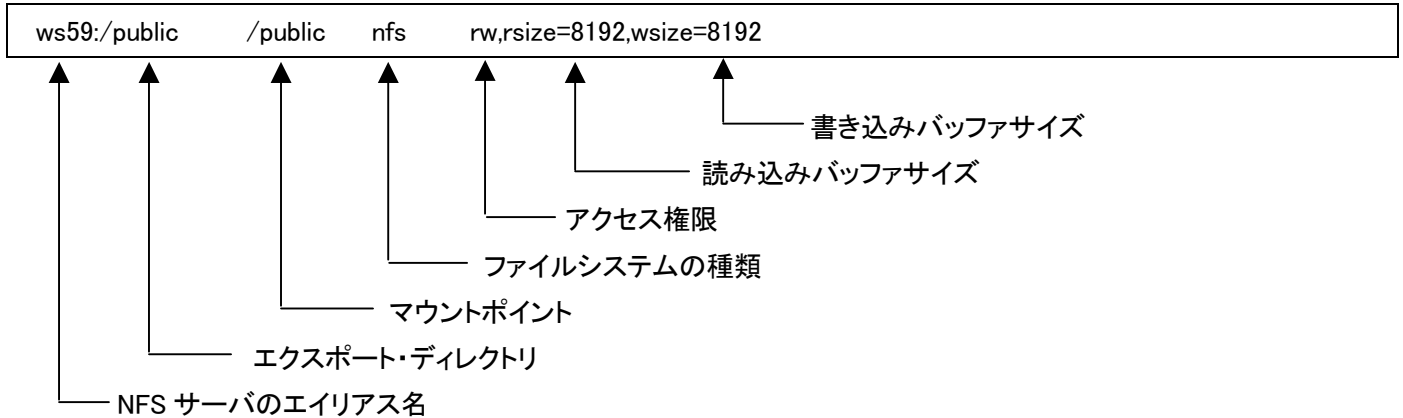
◇NFS マウントを終了するための umount コマンドの実行例

```
# umount /media
```

(7) NFS サーバへの自動マウントの設定

- NFS サーバへマウントするたびに、手作業で NFS マウントを行いたくない場合には、Linux の起動時に自動的に NFS マウントするように設定することができます。
- /etc/fstab ファイルの末尾に、次のような内容の記述を追加してください。

◇起動時に自動的に NFS マウントするための設定例



(8) exportfs コマンドによるエクスポート・テーブルの変更

/etc/exports ファイルの設定内容を変更したときに、次のようなステートメントを実行すると、NFS サーバを再起動することなく、変更内容をエクスポート・テーブルへ反映させることができます。

◇NFS サーバを再起動することなく、変更内容をエクスポート・テーブルに反映されるためのステートメント

```
# exportfs -rav
```

注:

NFS サーバ構築の詳細については、2008 年 2 月 29 日に実施されました、Linux ハンズオンセミナー「NFS よるファイルサーバを構築しよう！」の配布資料を参照してください。

以 上