

一歩進んだサーバー 構築・運用術

written by 仙石 浩明

第3回 ネーム・サーバー(後編)

ドメイン・ネーム・システム(DNS)は、インターネットの根幹を支えるシステムです。DNSを理解することはインターネット・サーバーの運用に欠かせません。連載第3回目の今回は、ネーム・サーバーの実際の構築法を紹介します。1台のマシン上で内部向けと外部向けのネーム・サーバーをセキュリティに配慮して構築する方法を解説します。



転職してそろそろ1カ月になります。引越して忙しかった先月に負けず劣らず忙しいこの1カ月でした。やりたいことは山ほどあるのに手がまるで足りない状況なのです。どんどん人を増やしたいのですが、優秀な人ほど転職には無関心で、転職雑誌に求人広告を出してもあまり効果が無いようです*1。

私は、某大手電機メーカーから転職したのですが、大企業からベンチャーへの転職は、いまだ比較的珍しい事例らしく、入社翌日いきなり某転職雑誌に取材されました。さらに、ベンチャー企業の取材をしているフリーの記者に、大企業から転職した人の話が聞きたいと取材を申し込まれ、転職を決意した経緯などについて1時間あまりお話ししました。

テレビの取材もありました。「NHKスペシャル」と「おはよ日本」の2つの撮影班が職場に来てカメラを回していましたので、私も映るかも知れません。また、今回の転職とは全く関係ないのですが、秋葉原で買い物をしていたらTBSテレビの「報道特集」のインタビューを受ける羽目になりました。米Palm社の日本進

出に関する特集番組を制作するとかで、Palm製品を購入した人の話を聞くということだったようです。もともと、私がそこで購入したのはTRGpro*2だったのですが。

というわけで、入社以来連続4週、取材を受けています。残念ながら来週は取材の予定が入っていないので、記録更新は止まりそうです。どなたか取材しに来ませんか?*3

2タイプのネーム・サーバー

先月号では、ネーム・サーバーの2つのタイプ、「サーバー専門型ネーム・サーバー」と「クライアント型ネーム・サーバー」について解説しました。サーバー専門型ネーム・サーバーとは、自分が知らないドメインに関する問い合わせに対しては、「ルート・サーバーに聞け」と冷たく言い放つタイプ、クライアント型ネーム・サーバーとは、自分が知らないドメインに関する問い合わせに対しても自ら調べてくれるタイプです。

先月号の結論をまとめると、

- ・組織外から自組織に関する問い合わせに答えるための、外向けサーバー専門型ネーム・サーバー
- ・組織内から全インターネットに関する問い合わせに答えるための、内向けクライアント型ネーム・サーバー

の2つのネーム・サーバーを立ち上げよ、ということになります。

マシンを何台も24時間稼働させておけるサイトならば、外向けネーム・サーバーを外部からアクセス可能なマシンで立ち上げ、内向けネーム・サーバーを外部からアクセスできないマシンで立ち上げれば良いでしょう。例えば、図1のようにルーターを2台用意してバリア・セグメントを構成します。

*1 興味ある方は、ぜひご応募ください。

*2 米Technology Resource Group社(TRG)によるPalm互換機。CF(コンパクト・フラッシュ)スロット搭載が特徴です。メモリー・カードやモデム・カードが使えます。私は普段32Mバイトのメモリー・カードを差していますが、手軽にバックアップ/リストアが行えるので、ハード・リセットが恐くありません。

*3 と、ここで書いても、この雑誌が出るころには記録更新は止まっていることでしょう。

要さいマシンは絶対に侵入されないように防備を固め、ごく一部の限られたサーバー・ソフトのみを立ち上げます。例えば、外部あるいはバリア・セグメント*上のマシンと、内部LAN上のマシンとの間の通信を中継するためのプロキシだけにサービスを限定するなどが考えられます。

内側のルーターは、要さいマシンと内部LANとの間の通信以外はすべて遮断するように設定します。つまりインターネットから内部LANにアクセスするには、一度要さいマシンを経由しなければなりません。

犠牲マシンは、要さいマシンに比べれば防備が緩やかなマシンで、外向けの各種サービスを行うサーバーを立ち上げるためのものです。外向けネーム・サーバーのほか、Webサーバーやメール・サーバーなどを立ち上げます。犠牲マシンから内部LANへの通信はルーターで遮断されるので、万が一犠牲マシンが侵入されても、被害が内部LANに及ぶのを防ぐことができます。

犠牲マシンを外向けのサービスごとに設置できればそれが理想ですが、個人でインターネットに接続する場合、何台もマシンを動かしておくわけにはいかないのが現状でしょう。第一、電気代が馬鹿になりませんし、狭い家だ

とマシンを設置している部屋で寝ることになるわけで、騒音も無視できません。

そこで本稿では、1台のマシンで外向けネーム・サーバーと、内向けネーム・サーバーを立ち上げる方法を説明します。連載第1回で紹介したように、GCDのゲートウェイはマシン1台だけで、しかもNICは1枚しかありません。このゲートウェイでGCDの内部向けと外部向けのネーム・サーバーを立ち上げています。

● 1台のマシンに
2つのネーム・サーバーを立ち上げる

ネーム・サーバーが利用するポートは53番と決まっています。そのため、ネーム・サーバーを2つ立ち上げるには、IPアドレスが2つ必要です。例えば私のマシン「asao.gcd.org」の場合、210.145.125.162と192.168.1.1の2つのIPアドレスを持っています。前者はインターネット全体で唯一のアドレスであるグローバル・アドレス、後者はサイト内部のみで使用可能なプライベート・アドレスです。

210.145.125.162のポート53番へアクセスすれば、外向けネーム・サーバーが応答し、192.168.1.1のポート53番へアクセスすれば、内向けネーム・サーバーが応答するように、そ

れぞれのネーム・サーバーを設定します。

このような場合、一般の解説書ではNIC (Network Interface Card) を2枚差して片方にグローバル・アドレスを割り当て、もう片方にプライベート・アドレスを割り当てると説明してあります。つまり図2のような構成です。

もちろんこのような構成でも問題無いのですが、なぜNICが2枚も必要なのでしょう。Linux など多くのOSで1つのNICに複数のIPアドレスを割り当てられます。つまり図3のような構成にすることも可能です。内部LANの各マシンと外部との直接通信は一切できないようにルーターにフィルタリングの設定をします。

図2と図3の構成は、セキュリティ上は五十歩百歩です。つまりゲートウェイが侵入されれば内部LANが危険にさらされます。NICを2枚入れたことによるアドバンテージは、全く無いと言っても過言ではないでしょう。

強いて言えば、図3のLANでは、グローバル・アドレスあてとプライベート・アドレスあての両方のパケットが流れますから、トラフィックが倍になります。マシンが何十台もつながっていれば問題になるかも知れませんが、家庭内LANではせいぜい数台のマシンしかつながないでしょうから、全く問題はありません。

図2のLANだと、内部LANにつないでいたマシンにグローバル・アドレスを割り当てようとしたとき、バリア・セグメントにマシンを接続し直す必要が生じますが、図3のLANだとマシンのIPアドレスを変更するだけで済みます。

家庭内LANでは、ケーブルの引き回しにいろいろと制約がある場合が多いので*4、ケーブルのつなぎ直しの必要がない図3の構成の方が圧倒的に便利と言えるでしょう。

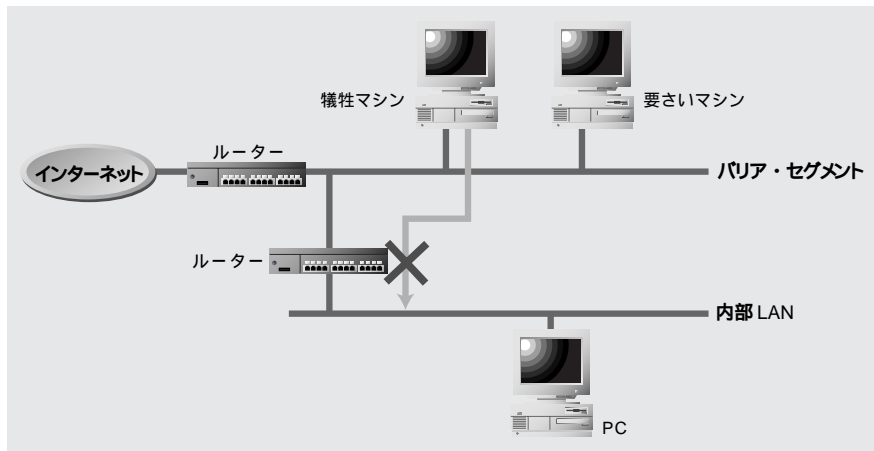


図1 バリア・セグメントの構成例
内部LAN上のマシンとインターネットとの間の通信は要さいマシンを経由してだけ行います。犠牲マシンは各種サーバーを立ち上げるためのものです。

【バリア・セグメント】 外部にサービスを提供しつつ、安全にLANをインターネットに接続するために、LANとインターネットの間に設けられたネットワーク・セグメントのことです。境界ネットワークなどと呼ばれる。

*4 内部LANとバリア・セグメントの2系統のケーブルを部屋中に張り巡らすのは、かなり大変でしょうね。

犠牲パーティション

外向けネーム・サーバーは、本質的に不特定多数に対してサービスを行う必要があります。したがって攻撃を受ける危険が高いサーバーの一つと言えます。万一侵入されても被害が限定できるようなサーバー構築が必要です。図1のような構成のネットワークであれば、サーバー・ソフトを犠牲マシン上で動かすことにより、万一侵入されても被害を犠牲マシンに限定することができます。

図3のような構成のネットワークの場合に、被害を限定するためのツールがchrootです。chrootを使うと、任意のディレクトリをルート・ディレクトリに設定してサーバー・プログラムを実行できます。そのプログラムは、設定されたルート・ディレクトリの外にあるディレクトリにアクセスすることはできません。もちろん、ルート・ディレクトリを元のルートに戻すことは不可能です(カーネルにバグがなければ)。

したがって、このサーバー・プログラムが侵入に利用されたとしても、侵入者が自由にアクセスできるのは、ルート・ディレクトリとして設定されたディレクトリの中だけで、もしこのディレクトリが、shやperlなど侵入者が喜びそうなものは一切置かないパーティションであれば被害をこのパーティション内に限定することができます。このようなパーティションを犠牲パーティションと呼びます(図4)。

bind-8.2.2-P5などの最近のネーム・サーバーでは、chrootの機能が内蔵されていますので、-tオプションに犠牲パーティションを指定するだけで済みます。例えば、犠牲パーティション「/alt-root」を作って、外向けネーム・サーバーを立ち上げるには、図5のように実行します。

「-u daemon -g daemon」は、namedをdaemon権限で動かすためのオプションです。ネーム・サーバーは立ち上げが完了してしま

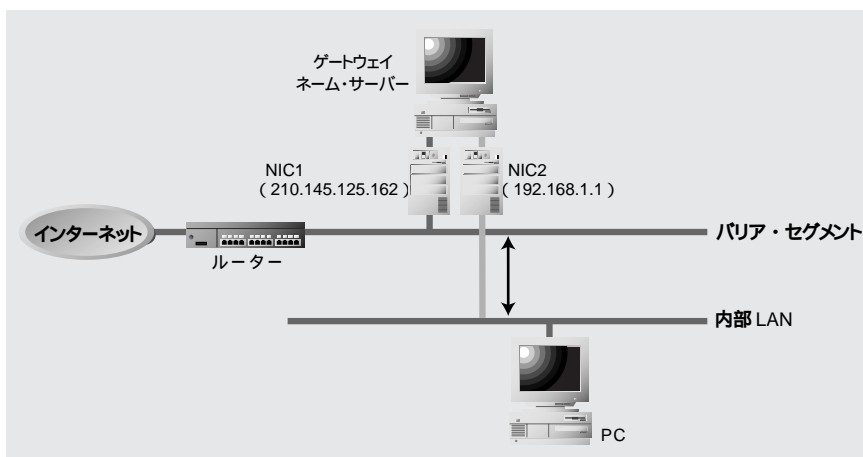


図2 NICを2枚使ったゲートウェイの構成例
ネットワークをセグメント分けする必要がありますので、家庭内LANのような環境だと無駄が多く、その割にセキュリティが向上するわけではありません。

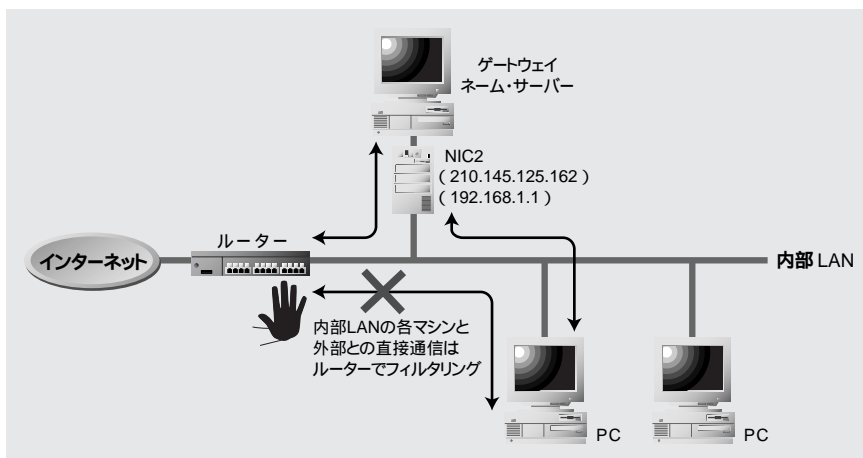


図3 NIC1枚だけでゲートウェイを構成
LANのトラフィックは多少増えますが、家庭で使用する程度のマシン数であれば問題ありません。

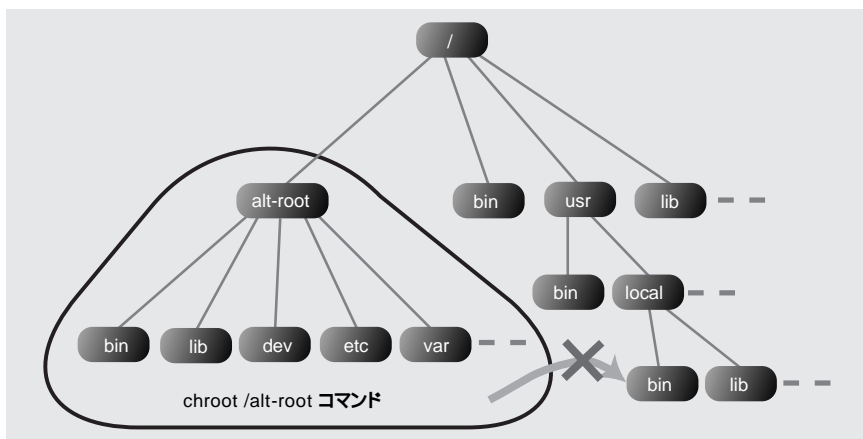


図4 犠牲パーティション
chrootを使ってルート・ディレクトリを移動しておくことで、侵入された場合の被害を限定することができます。

```
# named -t /alt-root -u daemon -g daemon
```

図5 犠牲パーティションを作って、外向けネーム・サーバーを立ち上げる方法
「-u daemon -g daemon」は、namedをdaemon権限で動かすためのオプションです。 このマークで改行

```

1 options {
2     directory "/var/named";
3     pid-file "/var/named/public.pid";
4     listen-on {
5         210.145.125.160/28;    // gcd.org
6     };
7     allow-transfer {
8         none;
9     };
10    fetch-glue no;
11    recursion no;
12 };
13
14 zone "gcd.org" {
15     type master;
16     file "db.gcd.org";
17     allow-transfer {
18         203.139.160.74;      // ns-tk012.ocn.ad.jp
19         210.167.164.35;     // brother.daionet.gr.jp
20     };
21 };
22
23 zone "160.125.145.210.IN-ADDR.ARPA" {
24     type master;
25     file "db.210.145.125.160";
26     allow-transfer {
27         203.139.160.74;      // ns-tk012.ocn.ad.jp
28     };
29 };
30
31 // master server
32
33 zone "haniwa.com" {
34     type master;
35     file "db.haniwa.com";
36     allow-transfer {
37         210.167.164.35;     // brother.daio.net
38     };
39 };
40
41 // slave server
42
43 zone "daio.net" {
44     type slave;
45     file "bak.daio.net";
46     masters {
47         210.167.164.35;     // brother.daio.net
48     };
49 };
50
51 // local
52
53 zone "0.0.127.IN-ADDR.ARPA" {
54     type master;
55     file "db.127.0.0";
56 };
57
58 zone "." {
59     type hint;
60     file "root.cache";
61 };

```

図6 外向けネーム・サーバーの設定ファイル/alt-root/etc/named.confの例

表1 外向けネーム・サーバーの実行に最低限必要なファイルやディレクトリ

/alt-root/etc/named.confの設定例を図6に挙げています。

ファイル名(ディレクトリ名)	内容
/alt-root/etc/named.conf	設定ファイル
/alt-root/var/named/	作業ディレクトリ
/alt-root/dev/log	ログの出力先(syslogdが作るソケット)

えば^{*5},root権限で動かす必要はありませんから,このようにroot以外の権限で走るようにしてしまえばより安全です。

犠牲パーティション/alt-rootには,外向けネーム・サーバーの実行に最低限必要なファイルやディレクトリ(表1)だけを整えておきます。ネーム・サーバーの実行ファイルやダイナミックライブラリも必要のように思えますが,これらは起動時にオープンされますから/alt-root下には置きません。/alt-root/var/named/のディレクトリは,設定ファイルnamed.confで変更可能です。

/alt-root/dev/logについては少々説明が必要でしょう。多くのサーバー・プログラムは,/dev/logへログ・データを出力します。/dev/logはunix-domainソケットと呼ばれる特殊ファイルです。サーバー・プログラムが出力したログ・データを,syslogdが/dev/logから読み込み,/etc/syslog.confの設定に従って/var/adm/syslog等のログ・ファイルへ出力します。

当然,/alt-rootへchrootしたネーム・サーバーは,/alt-root/dev/logにログ・データを出力しようとするから,syslogdは/dev/logと/alt-root/dev/logの両方からログ・データを読み込む必要があります。犠牲パーティションがほかにもある場合は,それぞれのパーティションの/dev/logを読み込まなければなりません。そのためには,次のように-aオプションを指定してsyslogdを実行します。-aオプ

*5 ポート53番を開くためにroot権限が必要なので,立ち上げ時にはrootで実行する必要があります。

シヨンは19個まで指定できます。

```
# syslogd -a /alt-root/dev/log
```

外向けネーム・サーバー

さて、ではいよいよnamedの設定ファイルの書き方の説明です。外向けのサーバー専門型ネーム・サーバーの設定ファイル/alt-root/etc/named.confの例を図6に示します。

1行目から12行目がnamedの全体的な設定、14行目以降がドメインごとの設定です。5行目でnamedが外向けのIPアドレスで問い合わせを待つように指定しています。8行目は、ゾーン転送をデフォルト禁止にする指定です。

重要なのは10行目と11行目で、この指定によりnamedがサーバー専門型ネーム・サーバーとして動作します。11行目の「recursion no」は文字通りrecursion(再帰)をしない、すなわち知らないドメインに関する問い合わせがあったときに他のサーバーへの問い合わせを行わないという指定です。

10行目は「糊」*6データを取り込まないという意味で、「recursion no」と組み合わせることにより他のネーム・サーバーが持っているドメインに関する情報をキャッシュに取り込むことを防ぎます。つまり「サーバー専門型」動作になります。

この例では、表2の各ドメインのプライマリ・サーバーになり、表3のドメインのセカンダリ・サーバーになります*7。

*6 ドメイン管理権限を委譲するときに、親ドメインと子ドメインを結び付けるためのデータを指します。具体的には、子ドメインのネーム・サーバーのホスト名とIPアドレスのことです。親ドメインが子ドメイン内に関する問い合わせを受け取ると、このデータを返します。

*7 実際のns.gcd.orgは、もっとたくさんのドメインを収容しています。

*8 このゾーン・ファイルは、執筆時点のgcd.orgのゾーン・ファイルそのままです。

【ゾーン・ファイル】ネーム・サーバーが情報に責任を持つ(権限を委譲されている)範囲を「ゾーン」と呼びます。ゾーンに関するデータを格納するのがゾーン・ファイルです。

ゾーン・ファイルは、named.confの2行目で指定したように、/alt-root/var/namedディレクトリの下に置きます。

ドメインgcd.orgのゾーン・ファイルdb.gcd.orgの例を図7に示します*8。

最後の行「\$INCLUDE member.gcd.org」で、ファイル「member.gcd.org」をインクルードしていますが、このファイルはgcd.orgのサブドメインについて記述したものです。このファイルも、/alt-root/var/namedディレクトリに置きます。一部を抜粋(図8)して紹介します。

db.210.145.125.160は逆引きのためのゾ

ン・ファイル*です。OCNエコノミーでは、ユーザーそれぞれに16個または8個のIPアドレスが割り当てられます。逆引きのためのゾーン・ファイルは、IPアドレス256個分が最小単位ですから、このままではユーザーのネーム・サー

表2 プライマリ・サーバーとして機能するドメイン

ドメイン	ゾーン・ファイル
gcd.org	db.gcd.org
160.125.145.210.in-addr.arpa	db.210.145.125.160
haniwa.com	db.haniwa.com
0.0.127.in-addr.arpa	db.127.0.0

表3 セカンダリ・サーバーとして機能するドメイン

ドメイン	ゾーン・ファイルのコピー
daio.net	bak.daio.net

```
; GCD.ORG.
;
$TTL 14400 ; Default TTL of 4 hour

@ IN SOA ns.gcd.org. sengoku.gcd.org. (
    58 ; Serial
    3600 ; Refresh after 1 hour
    1800 ; Retry after 30 min
    604800 ; Expire after 1 week
    3600 ) ; Minimum TTL of 1 hour
IN NS ns.gcd.org.
IN NS brother.daio.net.
IN NS ns-tk012.ocn.ad.jp.
IN A 210.145.125.162
IN MX 10 mx

localhost IN A 127.0.0.1
ns IN A 210.145.125.162
mx IN A 210.145.125.162
asao IN A 210.145.125.162
toyokawa IN A 210.145.125.163
soho IN A 210.145.125.168
soho1 IN A 210.145.125.169
soho2 IN A 210.145.125.170
soho3 IN A 210.145.125.171
soho4 IN A 210.145.125.172
ube IN A 210.145.125.174

www IN CNAME asao
news IN CNAME asao
uucp IN CNAME asao
mn IN MX 10 mx

$INCLUDE member.gcd.org
```

図7 ドメインgcd.orgのゾーン・ファイルdb.gcd.orgの例

```
maczuka IN MX 10 mx
*.maczuka IN MX 10 mx
www.maczuka IN CNAME www
```

図8 サブドメインについて記述したファイル「member.gcd.org」の一部だけを抜粋して紹介しています。

パーに逆引きのためのゾーン・ファイルを置くことができません。そこで、OCNではOCN側のネーム・サーバーで、図9のようなエイリアス（CNAME）が設定されています。

つまり、「161.125.145.210.in-addr.arpa. ~ 175.125.145.210.in-addr.arpa.」は、それぞれ「161.160.125.145.210.in-addr.arpa. ~

175.160.125.145.210.in-addr.arpa.」のエイリアス（別名）になっています。こうしておいて、ドメイン「160.125.145.210.in-addr.arpa.」に関する管理権限をユーザーのネーム・サーバー（この場合だとns.gcd.org）へ委譲しているわけです。ドメイン「160.125.145.210.in-addr.arpa.」

のゾーン・ファイル「db.210.145.125.160」の例を図10に示します。

例えば「210.145.125.162」の逆引きを調べようとして、「162.125.145.210.in-addr.arpa.」を問い合わせると、OCNのサーバーがCNAMEは「162.160.125.145.210.in-addr.arpa.」である旨を教えてくださいますから、

```
$ORIGIN 125.145.210.in-addr.arpa.

160  IN      NS      ns.gcd.org.
      IN      NS      ns-tk012.ocn.ad.jp.
161  IN      CNAME   161.160
162  IN      CNAME   162.160
163  IN      CNAME   163.160
164  IN      CNAME   164.160
165  IN      CNAME   165.160
166  IN      CNAME   166.160
167  IN      CNAME   167.160
168  IN      CNAME   168.160
169  IN      CNAME   169.160
170  IN      CNAME   170.160
171  IN      CNAME   171.160
172  IN      CNAME   172.160
173  IN      CNAME   173.160
174  IN      CNAME   174.160
175  IN      CNAME   175.160
```

図9 OCNのネーム・サーバーで設定されているエイリアス
ユーザーのネーム・サーバーに逆引きのためのゾーン・ファイルを置けるように、このようなエイリアス（CNAME）が設定されています。

```
; 160.125.145.210.IN-ADDR.ARPA.
;
$TTL      14400      ; Default TTL of 4 hour

@ IN SOA ns.gcd.org. sengoku.gcd.org. (
      5              ; Serial
      3600           ; Refresh after 1 hour
      1800           ; Retry after 30 min
      604800        ; Expire after 1 week
      1800          ; Minimum TTL of 30 min
      IN NS         ns.gcd.org.
      IN NS         ns-tk012.ocn.ad.jp.

; RFC 1101 stuff
0      IN PTR       gcdnet.gcd.org.
      IN A          255.255.255.240

162    IN PTR       asao.gcd.org.
163    IN PTR       toyokawa.gcd.org.
168    IN PTR       soho.gcd.org.
169    IN PTR       soho1.gcd.org.
170    IN PTR       soho2.gcd.org.
171    IN PTR       soho3.gcd.org.
172    IN PTR       soho4.gcd.org.
174    IN PTR       ube.gcd.org.137
```

```
1  options {
2      directory "/var/named";
3      pid-file "/var/named/private.pid";
4      listen-on {
5          192.168.1.0/24;      // gcd.org
6          127.0.0.1;
7      };
8      allow-query {
9          localnets;
10     };
11     allow-transfer {
12         none;
13     };
14     notify no;
15 };
16
17 zone "gcd.org" {
18     type master;
19     file "priv.gcd.org";
20 };
21
22 zone "160.125.145.210.IN-ADDR.ARPA" {
23     type master;
24     file "priv.210.145.125.160";
25 };
26
27 zone "1.168.192.IN-ADDR.ARPA" {
28     type master;
29     file "priv.192.168.1";
30 };
31
32 zone "localhost" {
33     type master;
34     file "priv.localhost";
35 };
36
37 zone "0.0.127.IN-ADDR.ARPA" {
38     type master;
39     file "db.127.0.0";
40 };
41
42 zone "." {
43     type hint;
44     file "root.cache";
45 };
```

図10 逆引き用ゾーン・ファイルの例
ドメイン「160.125.145.210.in-addr.arpa.」のゾーン・ファイル「db.210.145.125.160」の例です。

図11 内向けネーム・サーバーの設定ファイルの例

改めて「162.160.125.145.210.in-addr.arpa.」を問い合わせれば、「ns.gcd.orgが「asao.gcd.org」を返してくれます。

内向けネーム・サーバー

次に、内向けのクライアント型ネーム・サーバーの設定ファイルの例です(図11)。

```

; GCD.ORG. (private)
;
$TTL      86400      ; Default TTL of 1 day

@ IN SOA  ns.gcd.org.  sengoku.gcd.org.  (
    20      ; Serial
    3600    ; Refresh after 1 hour
    1800    ; Retry after 30 min
    604800 ; Expire after 1 week
    3600 ) ; Minimum TTL of 1 hour
IN NS     ns-local
IN A      192.168.1.1
IN MX     10 mx

localhost IN A      127.0.0.1
ns-local  IN A      192.168.1.1
mx        IN A      192.168.1.1
ns        IN A      210.145.125.162

www       IN CNAME   asaogw
news     IN CNAME   asaogt
uucp     IN CNAME   asaogw
mn       IN MX      10 mx

mucho    IN A      210.145.125.161
ns       IN A      210.145.125.162
asaogw   IN A      210.145.125.162
toyokawagw IN A      210.145.125.163
soho     IN A      210.145.125.168
soho1    IN A      210.145.125.169
soho2    IN A      210.145.125.170
soho3    IN A      210.145.125.171
soho4    IN A      210.145.125.172
ube      IN A      210.145.125.174
asao     IN A      192.168.1.1
asaogt   IN A      192.168.1.1
toyokawa IN A      192.168.1.2
toyokawagt IN A      192.168.1.2
kotohira IN A      192.168.1.3
ozenji   IN A      192.168.1.4
kawasaki IN A      192.168.1.6
mino     IN A      192.168.1.7
suited   IN A      192.168.1.9

$INCLUDE member.gcd.org
    
```

図12 ドメインgcd.orgのゾーン・ファイルpriv.gcd.orgの設定
外向けと内向けのネーム・サーバーで異なる結果を返せるのが分かります。

5行目と6行目でnamedが内向けのIPアドレスで問い合わせを待つように指定しています。表4のドメインについては外向けのネーム・サーバーと内向けのネーム・サーバーの両方でゾーン・ファイルを持っています。

したがって、これらのドメインに関する問い合わせに対しては、両者のネーム・サーバーで異なる結果を返すことができます。実際、ド

メインgcd.orgのゾーン・ファイルpriv.gcd.orgは、図12のようになっています*9。

kotohira, ozenji, kawasaki, mino, suitedはプライベート・アドレスしか持っていない、内部LANに接続されたホストです。当然、インターネットから直接アクセスすることはできませんから、外向けネーム・サーバーには登録していません。

一方, asaoとtoyokawaはグローバル・アドレスとプライベート・アドレスの両方を持っています。外向けネーム・サーバーにはグローバル・アドレスを登録し、内向けネーム・サーバーにはプライベート・アドレスを登録しています。また、内部からアクセスするときはグローバル・アドレスとプライベート・アドレスの両方が使え

*9 長くなるので、一部省略しています。

表4 管理権限の重なるドメイン

ドメイン	ゾーン・ファイル
gcd.org	priv.gcd.org
160.125.145.210.in-addr.arpa	priv.210.145.125.160

```

; 160.125.145.210.IN-ADDR.ARPA. (private)
;
$TTL      86400      ; Default TTL of 1 day

@ IN SOA  ns.gcd.org.  sengoku.gcd.org.  (
    2      ; Serial
    3600  ; Refresh after 1 hour
    1800  ; Retry after 30 min
    604800 ; Expire after 1 week
    1800 ) ; Minimum TTL of 30 min
IN NS     ns.gcd.org.
IN NS     ns-tk012.ocn.ad.jp.

; RFC 1101 stuff
0 IN PTR  gcdnet.gcd.org.
  IN A    255.255.255.240

161 IN PTR  mucho.gcd.org.
162 IN PTR  asaogw.gcd.org.
163 IN PTR  toyokawagw.gcd.org.
168 IN PTR  soho.gcd.org.
169 IN PTR  soho1.gcd.org.
170 IN PTR  soho2.gcd.org.
171 IN PTR  soho3.gcd.org.
172 IN PTR  soho4.gcd.org.
174 IN PTR  ube.gcd.org.
    
```

図13 priv.210.145.125.160
グローバル・アドレスに対応するホスト名にgwをつけることにより逆引きと正引きが一致するように設定しています。

ますから、両者を使い分けるために末尾にgtとgwをつけたホスト名を登録してあります。前者がプライベート・アドレスのホスト名、後者がグローバル・アドレスのホスト名です。

逆引きもこれに合わせて、外向けネーム・サーバーとはゾーン・ファイルを変える必要があります。図13のように、asaoとtoyokawaについては、グローバル・アドレスに対応するホスト名にgwをつけることにより逆引きと正引きが一致します。つまり、210.145.125.162を検索すると「asaogw」が得られ、「asaogw」を検索すると元の210.145.125.162が得られます。

動作確認

内向けのネーム・サーバー(asaogt)が、任意のドメインに関する問い合わせに対して正しく返答するか確認します(図14)。

一方、外向けのネーム・サーバー(asaogw)では、知っているドメインに関しては検索結果を答え(図15)、知らないドメインに関しては、「ルート・サーバーに聞け」と冷たく言い放つ(図16)ことを確認します。

外向けサーバーを何日間か走らせた後、キャッシュに他のドメインに関する情報を取り込

んでいないか、ネーム・サーバーにINTシグナルを送ってキャッシュの内容を出力させて調べます(図17)。/alt-root/etc/named.confの3行目で設定したように、/alt-root/var/named/public.pidには外向けネーム・サーバーのプロセスIDが入っています。このIDに対してkillコマンドでINTシグナルを送ると、/alt-root/var/named/named_dump.db^{*10}にネーム・サーバーが知っているドメインに関する情報と、キャッシュに記憶したデータが出力されます。見慣れないドメインに関するデータを取り込んでいないか確認します。

*10 ネーム・サーバーのコンパイル時の_PATH_DUMPFILE (デフォルトはnamed_dump.db)の設定に依存します。

```
% nslookup -q=any cybird.co.jp. asaogt.gcd.org
Server: asao.gcd.org
Address: 192.168.1.1

Non-authoritative answer:
cybird.co.jp      nameserver = dns1.cybird.co.jp
cybird.co.jp      nameserver = ns.ipro.ad.jp

Authoritative answers can be found from:
cybird.co.jp      nameserver = dns1.cybird.co.jp
cybird.co.jp      nameserver = ns.ipro.ad.jp
dns1.cybird.co.jp internet address = 210.156.250.253
ns.ipro.ad.jp     internet address = 203.179.10.4
```

図14 内向きネーム・サーバーの動作確認

```
% nslookup -q=any daio.net. asaogw.gcd.org
Server: asao.gcd.org
Address: 210.145.125.162
Aliases: 162.125.145.210.in-addr.arpa

daio.net      preference = 10, mail exchanger =
daio.daionet.gr.jp
daio.net      preference = 100, mail exchanger =
brother.daionet.gr.jp
daio.net      nameserver = brother.daio.net
daio.net      nameserver = ns.gcd.org
daio.net
    origin = brother.daio.net
    mail addr = postmaster.daio.net
    serial = 9
    refresh = 10800 (3H)
    retry = 3600 (1H)
    expire = 604800 (1W)
    minimum ttl = 86400 (1D)
daio.net      nameserver = brother.daio.net
daio.net      nameserver = ns.gcd.org
daio.daionet.gr.jp      internet address = 210.167.164.34
brother.daionet.gr.jp  internet address = 210.167.164.35
brother.daio.net      internet address = 210.167.164.35
ns.gcd.org            internet address = 210.145.125.162
```

図15 外向きネーム・サーバーの動作確認(1)
知っているドメインについては検索結果を答えます。

```
% nslookup -q=any cybird.co.jp. asaogw.gcd.org
Server: asao.gcd.org
Address: 210.145.125.162
Aliases: 162.125.145.210.in-addr.arpa

Authoritative answers can be found from:
(root) nameserver = J.ROOT-SERVERS.NET
(root) nameserver = K.ROOT-SERVERS.NET
(root) nameserver = L.ROOT-SERVERS.NET
(root) nameserver = M.ROOT-SERVERS.NET
(root) nameserver = I.ROOT-SERVERS.NET
(root) nameserver = E.ROOT-SERVERS.NET
(root) nameserver = D.ROOT-SERVERS.NET
(root) nameserver = A.ROOT-SERVERS.NET
(root) nameserver = H.ROOT-SERVERS.NET
(root) nameserver = C.ROOT-SERVERS.NET
(root) nameserver = G.ROOT-SERVERS.NET
(root) nameserver = F.ROOT-SERVERS.NET
(root) nameserver = B.ROOT-SERVERS.NET
J.ROOT-SERVERS.NET internet address = 198.41.0.10
K.ROOT-SERVERS.NET internet address = 193.0.14.129
L.ROOT-SERVERS.NET internet address = 198.32.64.12
M.ROOT-SERVERS.NET internet address = 202.12.27.33
I.ROOT-SERVERS.NET internet address = 192.36.148.17
E.ROOT-SERVERS.NET internet address = 192.203.230.10
D.ROOT-SERVERS.NET internet address = 128.8.10.90
A.ROOT-SERVERS.NET internet address = 198.41.0.4
H.ROOT-SERVERS.NET internet address = 128.63.2.53
C.ROOT-SERVERS.NET internet address = 192.33.4.12
G.ROOT-SERVERS.NET internet address = 192.112.36.4
F.ROOT-SERVERS.NET internet address = 192.5.5.241
B.ROOT-SERVERS.NET internet address = 128.9.0.107
```

図16 外向きネーム・サーバーの動作確認(2)
知らないドメインについては、自分で調べには行きません。

```
# kill -INT `cat /alt-root/var/named/public.pid`
```

図17 キャッシュの確認
ネーム・サーバーにINTシグナルを送ると、キャッシュに記憶したデータを調べることができます。