DNS完全解説 の仕組み

第11回 ネームサーバの適切な設定例

馬場達也

前回は、ドメイン名に日本語などの非ASCII文字を使用することが可能となる国際化ドメイン名の仕組みと利用法について解説した。今回は、これまでの復習として、ネームサーバを適切に運用するための設定例を紹介する。

実際にネームサーバを構築してみよう

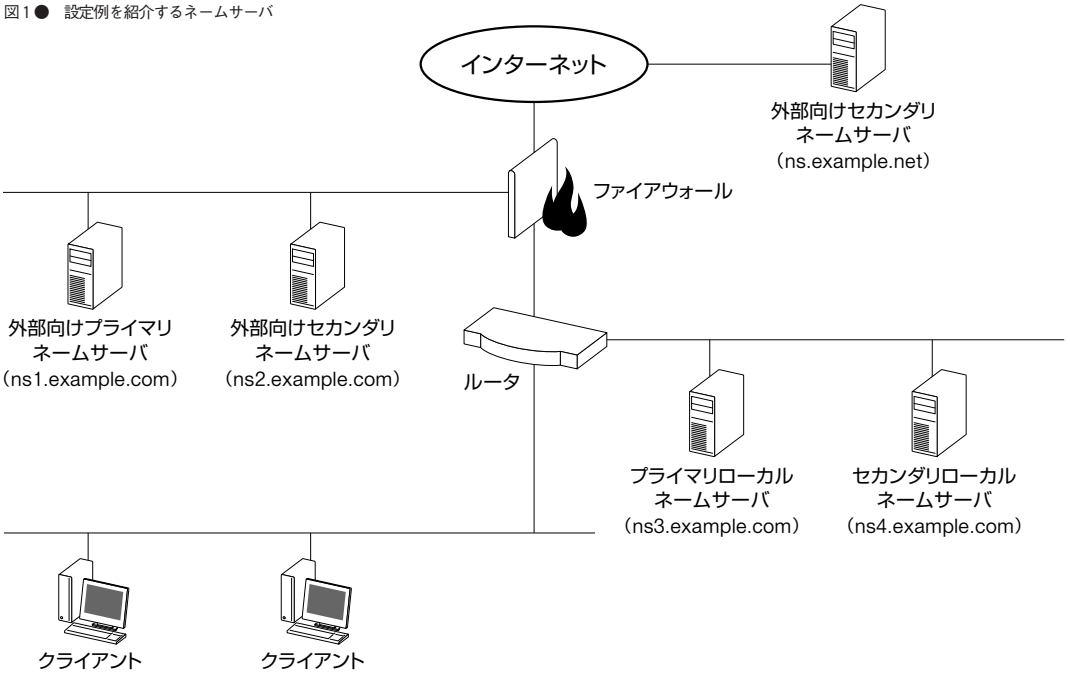
これまでの10回にわたる連載を通して、DNSのさまざまな機能とその仕組みについて解説してきた。しかし、その機能をどのように実際のネームサーバに適用すればよいかということについては、十分に触れていなかった。そこで今回は、ネームサーバの用途ごとに適切な設定例を紹介する。今回紹介するのは、図1に示すような、外部向けプライマリネームサーバ(ns1.example.com)、外部向けセカンダリネームサーバ(ns2.example.com)、そして、ローカルネームサーバ(ns3.example.com)の3種類のネームサーバの設定例である。

なお、ここで紹介する設定例は、BIND 9を基にしている。ほかのネームサーバを使用している場合でも基本的に考え方は同じなので、ここで示す設定例を参考にして自身のネームサーバを設定してもらいたい。

外部向けプライマリネームサーバを構築する

まず、BIND 9を用いて外部向けプライマリネームサーバを構築する際の設定例について紹介する。プライマリネームサーバは、あるゾーンのゾーンデータファイルを管理する権威ネームサーバである。権威ネームサーバの設定については、ルートネームサーバの運用条件について記述されているRFC 2870の内容を参考にすることができる。RFC 2870では、次のような条件が記述されている。

図1● 設定例を紹介するネームサーバ



【ハードウェア／ネットワークに関する条件】

●通常のピーク時の3倍のリクエストに耐えられるようにしておくこと

●インターネットとの十分な接続性を有すること

【ネームサーバ以外の設定に関する条件】

●UDPのチェックサム機能を使用すること

●DNSサービス以外のサービスを提供しないこと

●正確な時刻をログに記述するため、NTPで時刻を合わせておくこと

【ネームサーバの設定に関する条件】

●負荷を軽減し、ほかのネームサーバのデータをキャッシュしないようにするため、再帰問い合わせを受け付けないようにすること

●負荷を軽減するため、セカンダリネームサーバ以外からのゾーン転送のリクエストを受け付けないこと

●ネームサーバのホスト名とIPアドレスが正引きと逆引きで一致するようにすること

●TSIGなどを使用して、相手認証とメッセージ認証を行うこと

●メモリを節約するため、問い合わせを受けるゾーン以外のデータは保持しないこと

ここでは、ネームサーバの設定に関する条件にあげられているものを参考に、BIND 9での設定例を紹介する。ハードウェア／ネットワークに関する条件や、ネームサーバ以外の設定に関する条件についてはここでは解説しないが、ネームサーバを運用する際には重要な事項であるので、参考にしていただきたい。

最初に、外部向けプライマリネームサーバにおける「named.conf」ファイルの設定例をリスト1に示す。ここでは、まず、再帰問い合わせを受け付けないようにするために、optionsステートメントで「recursion no;」と設定する。これにより、権威ネームサーバ自身がほかのネームサーバに問い合わせを行わないようになり、その結果、キャッシュも行わなくなる。そして、インターネットで使用されていないアドレスから何らかのリクエストがあった場合には、そのリクエストに対して答えを返すのはむだであり、さらにそれが不正アクセスである可能性もあるため、このようなリクエストは無視するように設定する。この設定は、optionsステートメント内の「blackhole」サブステートメントで記述することができる。「blackhole」サブステートメント内でIPアドレスを記述すると、BIND 9は、そのアドレスからのリクエストを無視するようになる。ここでは、リスト1のように、プライベートアドレスや、予約アドレス、テストアドレスなどを指定しておく。

```
//オプション設定
options {
        directory "/var/named/";
        recursion no;                    ← 再帰問い合わせを受け付けない
        blackhole {                      ← 指定したアドレスからの問い合わせを無視する
                0.0.0.0/8;               ← 予約アドレス
                1.0.0.0/8;               ← 予約アドレス
                2.0.0.0/8;               ← 予約アドレス
                169.254.0.0/16           ← リンクローカルアドレス
                192.0.2.0/24;            ← テストアドレス
                224.0.0.0/3;             ← マルチキャストアドレス
                10.0.0.0/8;              ← プライベートアドレス
                172.16.0.0/12;           ← プライベートアドレス
                192.168.0.0/16;          ← プライベートアドレス
        };
};
// 正引きゾーン
zone "example.com" {
        type master;
        file "example.com.zone";
        allow-transfer { key axfr.example.com.; };   ← 指定したTSIG鍵を持つネームサーバからのゾーン転送のリクエストを許可
};
// 逆引きゾーン
zone  "0.135.163.in-addr.arpa" {
        type master;
        file "0.135.163.in-addr.arpa.zone";
        allow-transfer { key axfr.example.com.; };   ← 指定したTSIG鍵を持つネームサーバからのゾーン転送のリクエストを許可
};
// TSIG鍵の設定
key axfr.example.com. {
        algorithm hmac-md5;
        secret "nAgPY1cTo8HJ/FXw1waaow==";
};                                       ← ゾーン転送用のTSIG鍵を記述する
```

リスト1● 外部向けプライマリネームサーバにおける「named.conf」ファイルの設定例

ゾーン転送は、セカンダリネームサーバからのリクエストのみに答えるようにするために、zoneステートメント中の「allow-transfer」サブステートメントによって、ゾーン転送をTSIG鍵を持つセカンダリネームサーバのみに制限する。そして、keyステートメントでTSIG鍵を記述しておく。ISPなどのセカンダリDNSサービスなどを使用している場合などで、セカンダリネームサーバにおいてTSIGの設定を行うことが難しいケースでは、代わりに「allow-transfer { 163.135.0.11; };」のように、セカンダリネームサーバのIPアドレスを記述して、アドレスベースでゾーン転送を

リスト2● 正引き用ゾーンデータファイルの記述例（example.comゾーン）

```
$TTL 86400
@       IN      SOA     ns1.example.com.  hostmaster.example.com. (
                                2003041800 ; シリアル番号
                                28800      ; リフレッシュ間隔（秒）
                                7200       ; リトライ間隔（秒）
                                604800     ; ゾーンの有効期間（秒）
                                3600       ; ネガティブキャッシュの有効期間（秒）
                                )
        IN      NS      ns1.example.com.   ; このゾーンのプライマリマスタ
        IN      NS      ns2.example.com.   ; このゾーンのセカンダリマスタ
        IN      NS      ns.example.net.    ; このゾーンのセカンダリマスタ
        IN      MX      10 mx1.example.com.
        IN      MX      20 mx2.example.com.
        IN      A       163.135.0.30
ns1     IN      A       163.135.0.10
ns2     IN      A       163.135.0.11
mx1     IN      A       163.135.0.20
mx2     IN      A       163.135.0.21
www     IN      CNAME   example.com.
```

リスト3● 逆引き用ゾーンデータファイルの記述例（0.135.163.in-addr.arpaゾーン）

```
$TTL 86400
@       IN      SOA     ns1.example.com. hostmaster.example.com. (
                                2003041800 ; シリアル番号
                                28800      ; リフレッシュ間隔（秒）
                                7200       ; リトライ間隔（秒）
                                604800     ; ゾーンの有効期間（秒）
                                3600       ; ネガティブキャッシュの有効期間（秒）
                                )
        IN      NS      ns1.example.com.   ; このゾーンのプライマリマスタ
        IN      NS      ns2.example.com.   ; このゾーンのセカンダリマスタ
        IN      NS      ns.example.net.    ; このゾーンのセカンダリマスタ
10      IN      PTR     ns1.example.com.
11      IN      PTR     ns2.example.com.
20      IN      PTR     mx1.example.com.
21      IN      PTR     mx2.example.com.
30      IN      PTR     example.com.
```

制限するようにする。

また、ダイナミックアップデートを有効にすると、不正にゾーンを変更される可能性があるため、外部向けのネームサーバではダイナミックアップデートは許可しないようにする。ダイナミックアップデートは、デフォルトで許可していないので明示的に設定する必要はない。

正引き用のゾーンデータファイルは、リスト2のように記述する。このゾーンファイルのNSレコードには、上位ネームサーバに申請したネームサーバと同じものを記述する。また、NSレコードで記述したネームサーバが自ゾーンに属しているのであれば、そのネームサーバのAレコードもあわせて記述する。逆に、NSレコードで記述したネームサーバが外部ゾーンに属しているのであれば、このゾーン内にはそのネームサーバのAレコードを記述しないようにする。また、HINFO、TXT、WKSなどのレコードは、ホストに関する情報を攻撃者に与えてしまうことになるため、記述しないようにする。そして、NS、MX、CNAMEの各レコードの右辺に記述するホスト名には、CNAMEレコードで定義した別名ではなく、Aレコードで定義した正規名で記述する。このようなゾーンデータファイルを記述する際の注意点については、RFC 1912に記述されているので、参考にしてほしい。

次に、逆引き用のゾーンデータファイルをリスト3のように記述する。逆引き用ゾーンデータファイルで

は、正引き用ゾーンデータファイルで記述したAレコードに対する逆引き用レコード（PTRレコード）をすべて記述するようにする。

外部向けのセカンダリネームサーバを構築する

次に、BIND 9を用いてセカンダリネームサーバを構築する際の設定例を紹介する。セカンダリネームサーバの「named.conf」ファイルは、リスト4のように設定する。基本的にはプライマリネームサーバと同様に設定するが、正引きゾーンおよび逆引きゾーンに関するzoneステートメント内では「type slave;」と設定するところが異なる。

ほかのネームサーバからこのセカンダリネームサーバに対してゾーン転送のリクエストを行わないのであれば、「allow-transfer { none; };」と設定しておき、ゾーン転送のリクエストを受け付けないように設定す

リスト4● 外部向けセカンダリネームサーバにおけるnamed.confファイルの設定例

```
//オプション設定
options {
        directory "/var/named/";
        recursion no;                          再帰問い合わせを受け付けない
        blackhole {                            指定したアドレスからの問い合わせを無視する
                0.0.0.0/8;                     予約アドレス
                1.0.0.0/8;                     予約アドレス
                2.0.0.0/8;                     予約アドレス
                169.254.0.0/16                 リンクローカルアドレス
                192.0.2.0/24;                  テストアドレス
                224.0.0.0/3;                   マルチキャストアドレス
                10.0.0.0/8;                    プライベートアドレス
                172.16.0.0/12;                 プライベートアドレス
                192.168.0.0/16;                プライベートアドレス
        };
};
// 正引きゾーン
zone "example.com" {
        type slave;
        file "example.com.bak";
        masters { 163.135.0.10; };
        allow-transfer { none; };              ゾーン転送のリクエストを受け付けない
        notify no;                             NOTIFYメッセージを送信しない
};
// 逆引きゾーン
zone  "0.135.163.in-addr.arpa" {
        type slave;
        file "0.135.163.in-addr.arpa.bak";
        masters { 163.135.0.10; };
        allow-transfer { none; };              ゾーン転送のリクエストを受け付けない
        notify no;                             NOTIFYメッセージを送信しない
};
// TSIGを使用するサーバの設定
server 163.135.0.10 {                          プライマリネームサーバのIPアドレスを記述する
        keys { axfr.example.com.; };           ゾーン転送用のTSIG鍵を指定する
};
// TSIG鍵の設定
key axfr.example.com. {
        algorithm hmac-md5;
        secret "nAgPY1cTo8HJ/FXw1waaow==";     ゾーン転送用のTSIG鍵を指定する
};
```

る。また、この場合には、ほかのネームサーバに対してゾーンが変更されたことを知らせるNOTIFYメッセージを送信する必要がないため、zoneステートメント内で「notify no;」と記述して、NOTIFYメッセージを送信しないように設定する。そして、プライマリネームサーバでTSIGを使用したゾーン転送のリクエストのみを許可した場合には、serverステートメントおよびkeyステートメントにおいてTSIGの設定を行っておく。

BIND 9でローカルネームサーバを構築

ローカルネームサーバは、クライアントリゾルバからの再帰問い合わせを受け付けて、外部のネームサーバに対して代理で問い合わせを行う。また、ローカルネームサーバは、ローカルネットワークのゾーンを管理して、内部ネットワークにおける権威ネームサーバの役割を兼ねることが多い。ここでは、BIND 9を用いて、このようなローカルネームサーバを構築する場合の設定例を紹介する。ローカルネームサーバのnamed.confファイルはリスト5のように設定する。

最初に、ローカルネームサーバが悪意のある者に不正に利用されないように、問い合わせ元を制限する。これは、optionsステートメントの「allow- query」サブステートメントで、ローカルネームサーバを利用するネットワークのアドレスを記述することによって実現

リスト5● ローカルネームサーバにおけるnamed.confファイルの設定例

```
//オプション設定
options {
        directory "/var/named/";
        allow-query { 163.135.0.0/24; };          ← 問い合わせ元を制限する
};
//ルートネームサーバの設定
zone  "." {
        type hint;
        file "named.ca";
};
// 正引きゾーン
zone "example.com" {
        type master;
        file "example.com.zone";
        allow-transfer { key axfr.example.com.; };   ← 指定したTSIG鍵を持つネームサーバからのゾーン転送のリクエストを許可
        allow-update { key ddns.example.com.; };     ← 指定したTSIG鍵を持つクライアントからのダイナミックアップデートのリクエストを許可
};
// 逆引きゾーン
zone  "0.135.163.in-addr.arpa" {
        type master;
        file "0.135.163.in-addr.arpa.zone";
        allow-transfer { key axfr.example.com.; };   ← 指定したTSIG鍵を持つネームサーバからのゾーン転送のリクエストを許可
        allow-update { key ddns.example.com.; };     ← 指定したTSIG鍵を持つクライアントからのダイナミックアップデートのリクエストを許可
};
//ループバック用正引きゾーンの設定
zone  "localhost" {
        type master;
        file "localhost.zone";
        notify no;
};
//IPv4ループバックアドレス用逆引きゾーンの設定
zone  "0.0.127.in-addr.arpa" {
        type master;
        file "0.0.127.in-addr.arpa.zone";
        notify no;
};
```

リスト5続き

```
//IPv6ループバックアドレス用逆引きゾーンの設定（ip6.arpa）
zone "0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.ip6.arpa" {
        type master;
        file
"0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.ip6.arpa.zone";
        notify no;
};
//IPv6ループバックアドレス用逆引きゾーンの設定（ip6.int）
zone "0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.ip6.int" {
        type master;
        file
"0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.ip6.int.zone";
        notify no;
};
//プライベートアドレス用疑似ゾーンの設定
zone "10.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "16.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "17.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "18.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "19.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "20.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "21.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "22.172.in-addr.arpa" {
        type master;
```

リスト5続き

```
        file "dummy.zone";
        notify no;
};
zone "23.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "24.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "25.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "26.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "27.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "28.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "29.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "30.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "31.172.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
zone "168.192.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
```

リスト5続き

```
};
zone "254.169.in-addr.arpa" {
        type master;
        file "dummy.zone";
        notify no;
};
// TSIG鍵の設定
key axfr.example.com. {
        algorithm hmac-md5;
        secret "nAgPY1cTo8HJ/FXw1waaow==";   ← ゾーン転送用のTSIG鍵を記述する
};
key ddns.example.com. {
        algorithm hmac-md5;
        secret "iSUuKtizDJEK/9ptgewTHQ==";   ← ダイナミックアップデート用のTSIG鍵を記述する
};
```

リスト6● 疑似ゾーンの設定内容

```
$TTL 86400
@       IN      SOA     ns3.example.com.  hostmaster.example.com. (
                                2003041800 ; シリアル番号
                                28800      ; リフレッシュ間隔（秒）
                                7200       ; リトライ間隔（秒）
                                604800     ; ゾーンの有効期間（秒）
                                3600       ; ネガティブキャッシュの有効期間（秒）
                                )
        IN      NS      ns3.example.com.   ; 自身のホスト名
```

できる。これにより、指定した範囲のIPアドレスを持つクライアント以外からの問い合わせを受け付けないようになる。

そして、ローカルネームサーバでは、不必要な問い合わせが外部のネームサーバに発生しないように設定する。例えば、プライベートアドレスやループバックアドレスなどの、インターネット上で使用されていないアドレスに対する逆引き問い合わせが外部ネームサーバに対して発生しないように、これらのアドレスに対する逆引きゾーンをローカルネームサーバに疑似的に登録しておくようにする。こうすることで、クライアントからプライベートアドレスに対する逆引き問い合わせが発生した場合に、ルートサーバに対してむだな逆引き問い合わせが発生するのを防ぐことができる。この擬似的に追加した逆引きゾーンは、リスト6のように、SOAレコードと自身のネームサーバを記述したNSレコードを記述したゾーンデータファイルを用意しておくだけでよい。

また、ダイナミックアップデートを使用している環境では、各ゾーンのzoneステートメントにおいて、「allow-update」サブステートメントを使用して、ダイナミックアップデートを許可するように設定する。このサブステートメントでは、IPアドレスまたはTSIG鍵を指定できるが、許可するクライアントをIPアドレスで指定すると、IPアドレスを偽造された場合に簡単にゾーンデータが変更されてしまうため、TSIGで認証されたクライアントからのダイナミックアップデートのリクエストのみを許可するようにする。

今回は、外部向けプライマリネームサーバ、外部向けセカンダリネームサーバ、ローカルネームサーバのそれぞれについて、ネームサーバを適切に運用するための設定例を紹介した。次回は、今後のDNSの動向を紹介して、この連載の最終回としたい。

NTTデータ 馬場達也

●今回の内容に関連するRFC

RFC 1912 "Common DNS Operational and Configuration Errors"
RFC 2870 "Root Name Server Operational Requirements