DNS完全解説 の仕組み

第4回 メール配送におけるDNSの役割とDNSによるサーバの探索

馬場達也

前回は、DNSの逆引きの仕組みと、逆引き用のゾーンデータファイルの記述法について紹介した。今回は、メール配送におけるDNSの役割と、配送先メールサーバを指定するためのMXレコードの記述法、そして、ある特定のサービスを提供するサーバを探索するために使用されるSRVレコードについて紹介しよう。

メールの配送先を教えてくれるMXレコード

電子メールは、複数のメールサーバを経由して、目的のユーザーのメールボックスに届けられる。このとき、メールを中継するメールサーバは、次の配送先のメールサーバを宛先のメールアドレスから判断しなければならない。配送先が組織内のメールサーバであれば、次の配送先を静的に指定することも可能だが、ほかの組織のメールサーバへ配送する場合には静的に記述しておくことは難しい。このような場合は、DNSを使用して配送先のメールサーバを検索することができる。DNSには「MXレコード」というリソースレコードがあり、メールアドレスの「@」よりもうしろのドメイン名をキーにしてMXレコードを引く(検索する)ことにより配送先のメールサーバを知ることができる。

MXレコード

MX(Mail Exchange)レコードは、メールの配送先となるメールサーバを記述するためのリソースレコードである。MXレコードの書式はRFC 1035で次のように規定されている。

```
<owner> <ttl> <class> MX <preference> <exchange-dname>
```

<owner>には、宛先のメールアドレスの「@」よりもうしろのドメイン名を記述する。<preference>には、配送の優先度を示すプリファレンス値(値が小さいほど優先される)を記述する。また、<exchange-dname>には、配送先のメールサーバの名前を正規名で記述する。<ttl>には、このリソースレコードのキャッシュの有効期間を秒単位で記述する。<ttl>を省略した場合には、直前の$TTL制御ステートメントでセットされたデフォルトの有効期間がセットされる。<class>には、ネットワーククラスを記述し、インターネットでは「IN」と記述する。<class>を省略した場合には、自動的に「IN」がセットされる。MXレコードの記述例を次に示す。この例では、宛先メールアドレスが「~@example.com」であるメールの配送先が「mx1.example.com」であることを示している。

```
$ORIGIN example.com.
@    IN    MX    10  mx1.example.com.
```

では、ここで、MXレコードを利用した電子メールの配送手順について説明しよう。まず、メールを受信したメールサーバは、メールのエンベロープヘッダの「RCPT TO:」フィールドの宛先メールアドレスを確認する(エンベロープヘッダとは、SMTPで使用される配送のための情報であって、メールメッセージに付加される「To:」ヘッダとは異なる)。例えば、この宛先メールアドレスが「baba@example.com」となっていれば、配送元のメールサーバは、DNSで「example.com.」をキーにしてMXレコードを引く。すると、example.comゾーンを管理するネームサーバは、「example.com.」のMXレコードと、そのMXレコードに含まれているネームサーバのAレコードを返却する(図1)。もし、Aレコードが返却されずに、MXレコードだけが返却された場合は、配送先メールサーバのIPアドレスを知るために、さらにAレコードを引き直す。このようにして、メールサーバは、配送先のメールサーバのホスト名とIPアドレスを得るのである。

メールサーバの配送の優先度

もし、システム障害やネットワーク障害などでメールの配送に失敗した場合は、配送元組織のメールサー

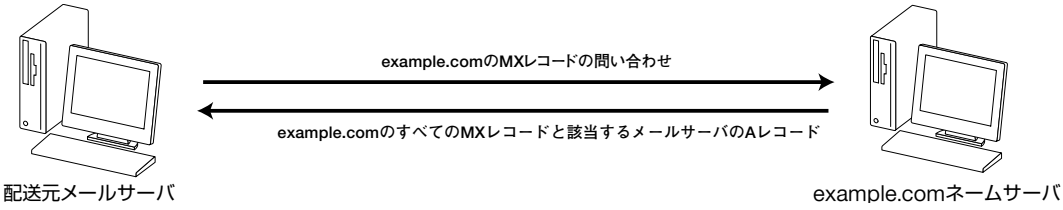


図1● MXレコードを使って配送先メールサーバを検索する

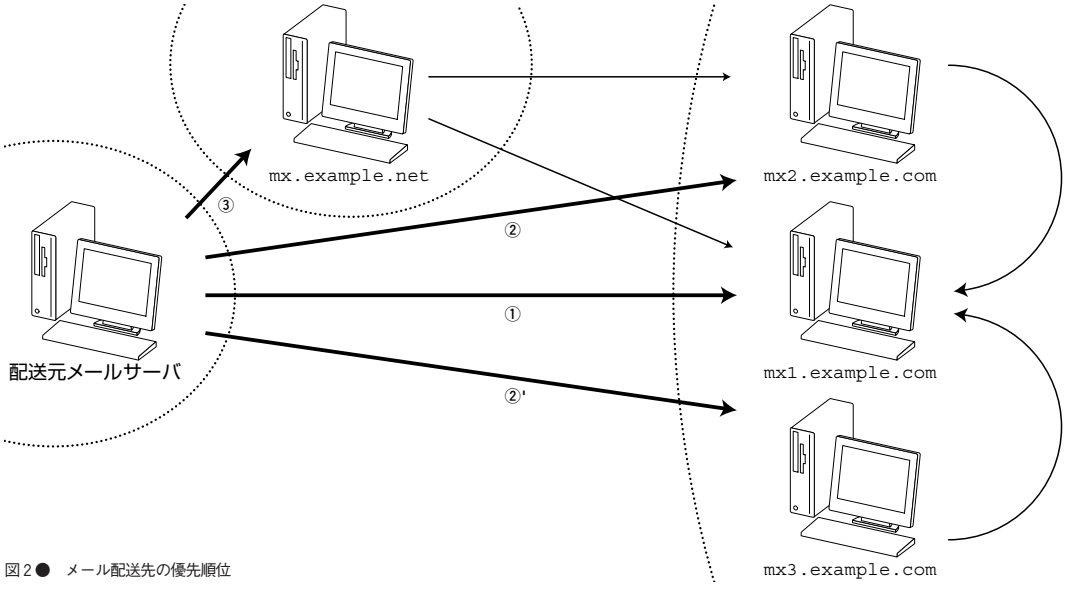


図2● メール配送先の優先順位

```
$ORIGIN example.com.
@      IN    MX    10  mx1.example.com.   ← プライマリメールサーバ
       IN    MX    20  mx2.example.com.   ← セカンダリメールサーバ
       IN    MX    20  mx3.example.com.   ← セカンダリメールサーバ
       IN    MX    30  mx.example.net.    ← セカンダリメールサーバ
```

リスト1● プリファレンス値による配送先の優先度の指定

バは、メールを送れるようになるまで待っていなければならない。このため、通常はメールを受け取るためのメールサーバを複数用意しておき、主としてメールを受け取るメールサーバ（これを「プライマリメールサーバ」という）がダウンしていた場合は、ほかのメールサーバ（これを「セカンダリメールサーバ」という）がメールを受け取れるようにしておく。この場合のメールの配送先の優先順位は、MXレコードのプリファレンス値で指定する。つまり、プライマリメールサーバには、最も小さいプリファレンス値を付けておき、セカンダリメールサーバには、プライマリネームサーバよりも大きいプリファレンス値を付けておく。セカンダリメールサーバの間では、プライマリメールサーバに近い（例えば、同じセグメント上の）セカンダリメールサーバから、小さいプリファレンス値を付けていくようにする。

例えば、リスト1のように、MXレコードが記述されていた場合には、通常はプリファレンス値が最も小さいmx1.example.comにメールを配送する（図2の①）。しかし、このメールサーバがダウンしていた場合には、次にプリファレンス値の小さいmx2.example.comまたはmx3.example.comのどちらかにメールを配送しようとする（図2の②および②'）。さらに、mx2.example.comとmx3.example.comの両方に対してもメールの配送に失敗した場合は、次にプリファレンス値の小さいmx.example.netにメールを配送しようとする（図2の③）。特に、ネットワーク障害が発生した場合は、そのネットワーク上のすべてのメールサ

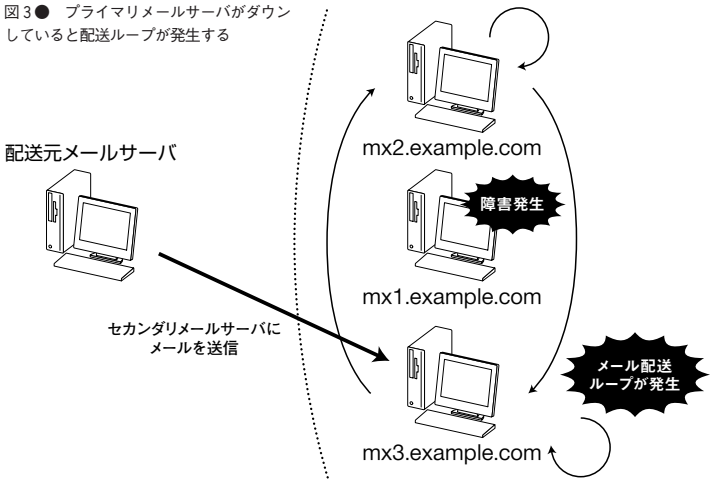
ーバに対してアクセスできなくなる場合も考えられるため、別のネットワーク上にもセカンダリメールサーバを設置しておくのがよいだろう。

では、プライマリメールサーバであるmx1.example.comがダウンしていて、セカンダリメールサーバであるmx2.example.comがメールを受け取った場合にはどうなるのだろうか。最終的には、プライマリメールサーバであるmx1.example.comにメールを配送しなければならないが、すぐにmx1.example.comにメールを配送しようとしても、まだ障害から復旧していないため配送に失敗してしまう。では、次にプリファレンス値の小さいmx2.example.comかmx3.example.comに配送するのであろうか。しかし、これらのメールサーバは、自分自身か、自分とプリファレンス値が同じメールサーバである。自分自身にメールを配送すると、メールプログラムがループを検出し、エラーとしてメールが返されてしまう。

また、自分とプリファレンス値が同じであるmx3.example.comに配送しても、配送先で同じ処理を行い、mx2.example.comとmx3.example.comの間でループが発生する可能性がある（図3）。自分よりプリファレンス値の大きいmx.example.netに送っても、プライマリメールサーバからさらに遠ざかるだけだ。

このため、メール配送プログラムは、取得したすべてのMXレコードのホスト名を確認し、その中に自分自身が含まれていれば、そのプリファレンス値を調べて、そのプリファレンス値と同じかさらに大きいプリファレンス値を持つMXレコードを無視するという処理を行う。こうすることによって、メールの配送先は、自分よりも小さいプリファレンス値を持つメールサーバのみに限定されるため、エラーの発生やむだな配送を防ぐことができるのである。

図3● プライマリメールサーバがダウンしていると配送ループが発生する



しかし、MXレコードで指定するメールサーバのホスト名に別名を使用していた場合は、メールを受け取ったメールサーバは、MXレコードの中に自ホストの名前を見つけられない可能性がある。この場合は、同じプリファレンス値のMXレコードを無視できず、メールの配送でループが発生してしまう可能性がある。このため、MXレコードで指定するメールサーバのホスト名は、必ず正規名で記述しなければならない。

MXレコードを記述する際の注意事項を整理する

ここでは、MXレコードを記述する際の注意事項について説明する。ここで紹介する注意事項については、RFC 1912とRFC 2181で詳しく述べられているので、こちらもぜひ参照していただきたい。

MXレコードで指定するメールサーバのホスト名は正規名でなければならない

先に説明したとおり、MXレコードに含まれるメールサーバのホスト名を別名（CNAMEで記述してあるホスト名）で指定すると、プライマリメールサーバがダウンしていた場合に、セカンダリメールサーバの間でメールの配送ループが発生してしまう可能性がある。さらに、DNSでMXレコードを検索した場合には、通常はMXレコードとともに、MXレコードに含まれるメールサーバのAレコードも返却されるが、多くの場合、CNAMEレコードまでは返却されない。このため、MXレコードに含まれるメールサーバのホスト名を別名で指定していた場合には、MXレコードしか返却されないことになり、配送元のメールサーバは、さらにMXレコードで返却されたメールサーバのホスト名の名前解決を行わなければならず、DNSの検索によけいな負荷と時間がかかってしまう。このため、MXレコードで指定するメールサーバのホスト名は、必ず正規名（Aレコードで記述した名前）で記述する。

［まちがった例］

```
@     IN   MX   10   mx1←  mx1はfooの別名である
mx1   IN   CNAME    foo
foo   IN   A    192.168.0.20
```

［正しい例］

```
@     IN   MX   10   foo←  正規名であるfooを記述する
foo   IN   A    192.168.0.20
```

ワイルドカードMXは使用しない

MXレコードでは「ワイルドカードMX」という一見便利そうな記述も可能だ。例えば、「～@～.example.com」というアドレスあてのメールをすべてmx1.example.comに配送させるように設定するためには、ワイルドカードを使用して次のように記述できる。

```
*.example.com.  IN  MX  10  mx1.example.com.
```

しかし、このように記述すると、存在しないドメイン名を含むメールアドレスあてのメールが送信された場合に、本来はDNSの検索でエラーとなるはずなのにワイルドカードMXにマッチするので、mx1.example.comがそのメールを受け取ってしまい、そのあとにメールをエラーとして返すというむだな処理が発生する。さらに、ワイルドカードは、検索したドメイン名に対してAレコードやNSレコードなどのほかのレコードが存在する場合には無視されるので、あまり有効に機能しない。このため、ワイルドカードMXは使用せず、MXレコードは、必要なぶんだけ正しく記述するようにする。

［まちがった例］

```
$ORIGIN example.com.
@    IN  MX  10 mx1.example.com.
*    IN  MX  10 mx1.example.com.
                ↑
          ワイルドカードMXを使用
```

［正しい例］

```
$ORIGIN example.com.
@    IN  MX  10 mx1.example.com.
sub1     IN  MX  10 mx1.example.com.
sub2     IN  MX  10 mx1.example.com.
sub3     IN  MX  10 mx1.example.com.
                    ↑
   メールアドレスに含まれるドメイン名のすべてに対して
   MXレコードを記述する
```

MXレコードを汎用的にしたSRVレコード

ある組織のメールサーバを探すにはDNSのMXレコードを引けばよいことは説明した。それでは、そのほかのサービスを提供するサーバを見つけるにはどうすればよいのだろうか。例えば、ある組織のWebサーバにアクセスする場合には、その組織のドメイン名に「www」を付けたものがWebサーバであると予測してアクセスするだろう。このように、ホスト名に、サービスの内容を連想させる別名を付けるということが一般的に行われている。この方法はRFC 2219にも記述されており、サーバには、提供するサービスに応じて、表1の別名を付けることが推奨されている。

別名	サーバの種類
archie	Archieサーバ
finger	Fingerサーバ
ftp	FTP（File Transfer Protocol）サーバ
gopher	Gopherサーバ
ldap	LDAP（Lightweight Directory Access Protocol）サーバ
mail	SMTP（Simple Mail Transfer Protocol）サーバ
news	NNTP（Network News Transfer Protocol）サーバ
ntp	NTP（Network Time Protocol）サーバ
ph	CCSO（Computing and Communications Services Office）ネームサーバ（Ph）
pop	POP（Post Office Protocol）サーバ
rwhois	RWHOIS（Referral WHOIS）サーバ
wais	WAIS（Wide Area Information Service）サーバ
whois	WHOISサーバ
www	Webサーバ

表1● サーバに対して付与する別名

しかし、これらの別名を付与するのは面倒な場合も多い。このため、DNSである特定のサービスを提供するサーバを発見するためのSRVレコードが提案されており、現在利用され始めている。

SRVレコード

MXレコードは、あるドメイン内のメールサーバのホスト名を取得するためのリソースレコードである。これに対して、SRV（Server Selection）レコードは、あるドメイン内で特定のサービスを提供しているサーバのホスト名を取得するためのリソースレコードであり、MXレコードを汎用的なものにして、ほかのサービスにも適用できるようにしたものと考えることができる。また、MXレコードと比較して、ロードバランシングの機能が強化されている。WindowsのActive Directoryでは、「ドメインコントローラ」やKerberosのKDC（Key Distribution Center）、LDAPサーバなどを探索する場合にこのSRVレコードを使用している。SRVレコードはRFC 2782に記述されており、次のような書式になっている。

```
<owner> <ttl> <class> SRV <priority> <weight> <port> <target>
```

<owner>には、「_サービス名._トランスポート層プロトコル名.ドメイン名」の形式で、あるドメインにおけるサービス名を記述する。例えば、example.comド

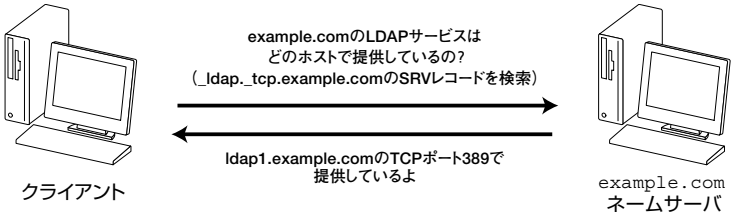


図4● SRVレコードを使ってサーバを発見する

```
$ORIGIN example.com.
_smtp._tcp  IN  SRV  10   0  25  mx1.example.com.
            IN  SRV  20   1  25  mx2.example.com.
            IN  SRV  20   4  25  mx3.example.com.
            IN  SRV  30   0  25  mx.example.net.
```

リスト2● SRVレコードによるメールサーバの指定

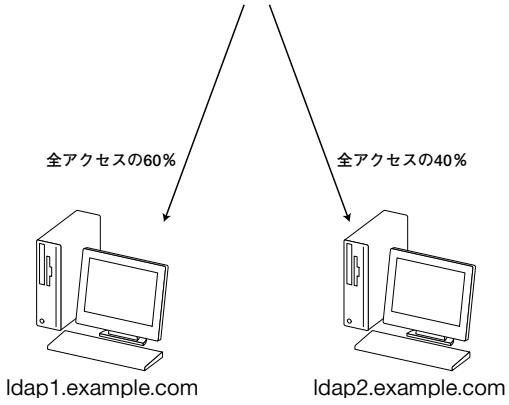


図5● SRVレコードを使ってロードバランシングを実現できる

メインのKerberosサービスであれば、「_kerberos._tcp.example.com.」のようになる。

<priority>は、MXレコードの<preference>と同じであり、<target>で記述したホストの優先度を記述する。この値が小さいほど優先度が高いことになる。<weight>には、優先度が同じサーバに対して、選択される確率の高さ（ロードバランシングの比重）を記述する。この値が大きいほど選択される確率が高くなる。ロードバランシングを行わない場合には「0」と記述する。

<port>には、そのホスト上でサービスを提供しているTCPまたはUDPのポート番号を記述する。<target>には、<owner>で記述したサービスを提供するサーバのホスト名を記述する。もし、サービスを提供するサーバが存在しないことを記述する場合には、このフィールドに「.」と記述する。

例えば、LDAPサーバのホスト名が「ldap1.example.com.」である場合には次のように記述する。これにより、「example.com内のLDAPサーバは？（_ldap._tcp.example.com.のSRVレコードは？）」という問い合わせに対して、LDAPサーバのホスト名（ldap1.example.com）とサービスを提供しているポート（389）を答えることができる（図4）。

```
$ORIGIN example.com.
_ldap._tcp  IN  SRV   0   0  389  ldap1
```

ロードバランシングを使用する場合は、次に示すように記述する。このように記述すると、図5のように、ldap1.example.comとldap2.example.comで6：4の割合でアクセスの負荷がかかるようにロードバランシングを実現できる。ロードバランシングを行うマシンで処理能力が異なる場合に有効である。

```
$ORIGIN example.com.

_ldap._tcp  IN  SRV   0   6  389  ldap1
            IN  SRV   0   4  389  ldap2
```

このSRVレコードは、リスト2のように記述すると、リスト1に示したMXレコードの代わりにもなる。当分はMXレコードが使われ続けるだろうが、そのうちにSRVレコードがMXレコードを置き換えるかもしれない。

次回は、ゾーン転送の仕組みについて説明する。

NTTデータ　馬場達也

●今回の内容に関連するRFC

RFC 1035 "Domain names - implementation and specification"
RFC 1912 "Common DNS Operational and Configuration Errors"
RFC 2181 "Clarifications to the DNS Specification"
RFC 2219 "Use of DNS Aliases for Network Services"
RFC 2782 "A DNS RR for specifying the location of services (DNS SRV)"
RFC 2821 "Simple Mail Transfer Protocol"