特集1 ネットワーク運用管理に必須

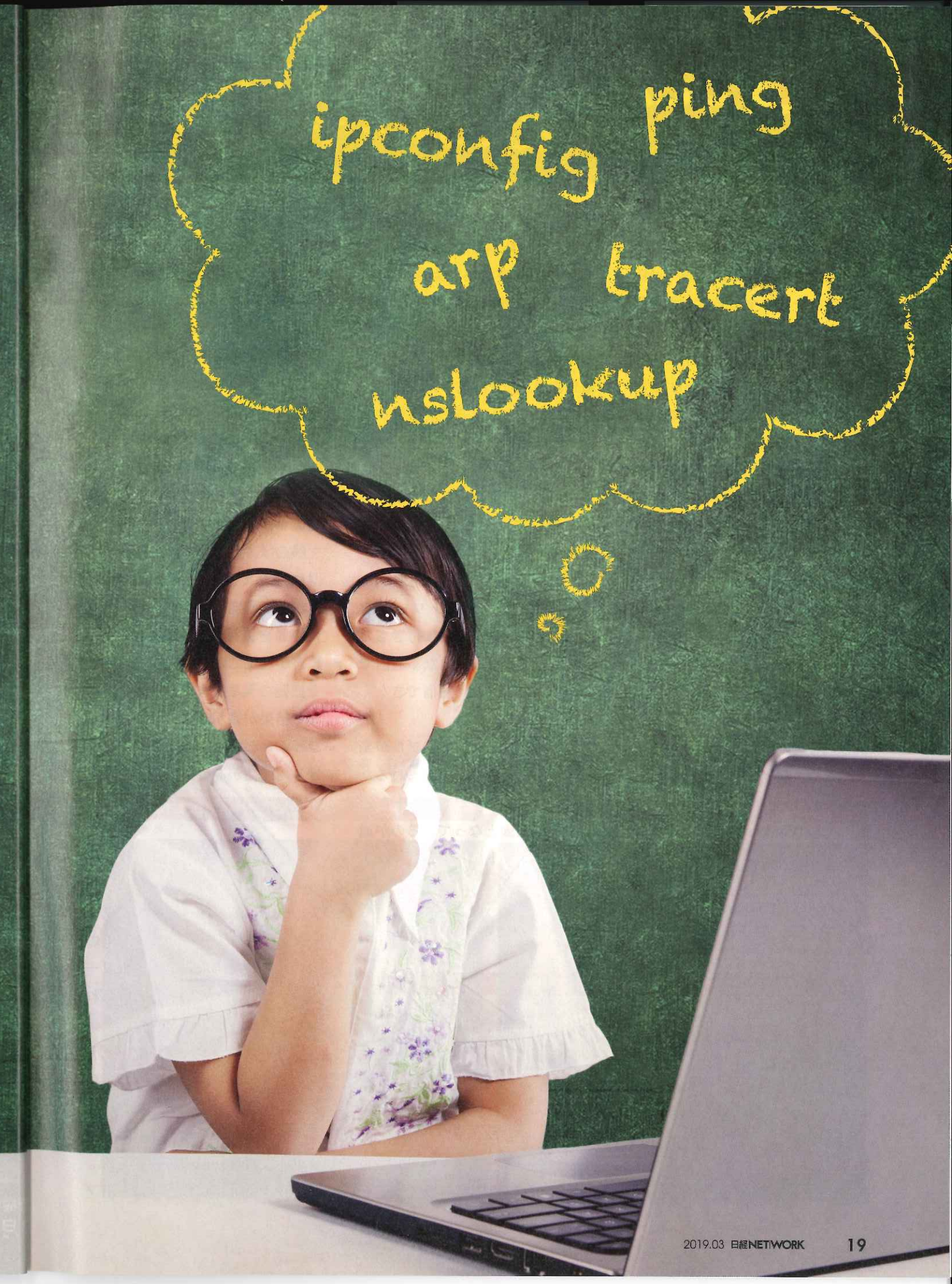
Windows 10時代のコマンド活用術

運用管理に欠かせないネットワークコマンドだが、Windows 10にはWindows 7/8.1と違う独特の“クセ”がある。把握していないと、思わぬ落とし穴にはまったり、せっかくの機能を十分に生かせなかったりする。この特集では、Windows 10が主役となった企業ネットワークにおけるネットワークコマンドの最新活用術を紹介する。（根本 浩之）

Part 1 ここが違うWindows 10コマンド

“クセ”を理解してコマンドでらくらく管理 ▶p.20

Part 2 知らなきゃ困る5大コマンド


ipconfig
ping
arp
tracert
nslookup

Part 1 ここが違うWindows 10コマンド
“クセ”を理解してコマンドでらくらく管理

企業のネットワーク管理者にとって必須の運用管理ツールが、ネットワークコマンドだ。ネットワークコマンドは、ネットワークの状況を調べたり、機器の状態を確認したりできる。しかもWindowsに標準で備わっているので、すぐに使える。例えば、ネットワークトラブルが発生したときに、手元に特別なツールがなくてもWindowsが動いているパソコンさえあれば、トラブルを解決することができる。

ただWindows標準のツールなので、Windowsのバージョンによって実行環境や使えるコマンドの種類も変わる。現在多くのユーザーが使っているWindows 7は、2020年1月にサポート期間が終了するため、最新のWindows 10に移行する必要がある。Windows 10には、Windows 7/8.1にない独特の“クセ”があるので要注意だ。

Part1では、ネットワークコマンドの基本的な使い方を解説する。さらにWindows 10のクセと、従来のユーザーがクセを気にせずに使えるようにするテクニックを紹介する。

便利なネットワークコマンド

まずは、ネットワークコマンドの利用シーンを説明しよう。典型的な使い方は、大きく2種類ある（**図1-1**）。

1つは、ネットワークトラブルの原因究明だ。ネットワークを使っているとトラブルは付きものだ。先ほどまでは問題なく通信できていたのに、突然通信できなくなることはよくある。

こうしたときに、トラブルが発生した原因をいち早く見つけるために、ネットワークコマンドが活躍する。原因となっているのが通信相手かネットワークか、ネットワークだとすればどの辺りが怪しいか、といったことを調べられる。ネットワークコマンドである程度のめどを立ててから、疑わしい箇所を詳細に調査すれば効率的にトラブルに対処できる。

もう1つは、日常的な動作監視だ。ネットワークコマンドを使えば、ネットワーク機器の状態を確認できるので、運用管理の効率化につながる。

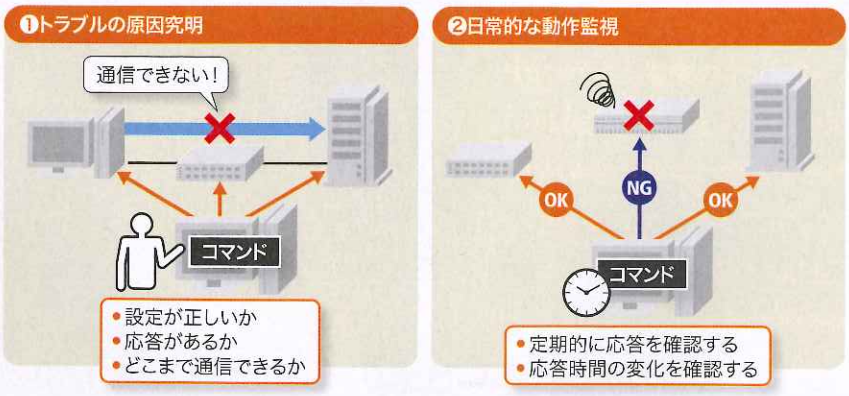
例えば、重要なネットワーク機器に対して、毎日決まった時間に状態を確認するネットワークコマンドを実行する。そうすれば、障害が発生した際に見逃すことなく把握できる。また、いつもと異なる結果が表示されれば調査するきっかけとなり、障害を未然に防げる可能性がある。

シェル上で実行する

Windowsに標準で備わっているネットワークコマンドだが、Windowsのスタートメニューを見てもネットワークコマンドそのものは確認できない。ネットワークコマンドは、「シェル」と呼ばれるプログラム上で実行する。シェルとは、ネットワークコマンドを含むOSの機能として実装しているコマンドを、ユーザーが利用しやすくするために用意されたプログラムである（**図1-2**）。

Windows 10には、コマンドプ

図1-1 運用管理に便利なネットワークコマンド
（1）トラブルの原因究明、（2）日常的な動作監視――などが可能。Windowsに限らずほとんどのOSに標準で含まれるのですぐに使える。



▼バッチファイル
コマンドプロンプトで打ち込む内容などを順番に記述したテキストファイル。プログラムと同じように実行できる。

▼パイプ
コマンドとコマンドの間で実行結果を受け渡す機能。

ロンプトとWindows PowerShell（以下、PowerShell）という2種類のシェルが用意されている（**図1-3**）。見た目に大きな違いはないが、機能に差がある。

コマンドプロンプトは、Windowsの前身であるMS-DOSと呼ばれるOSのユーザーインターフェースを引き継いだシェルだ。コマンドプロンプトでは、実行したいネットワークコマンドを文字で打ち込んで、Enterキーを押すと処理が実行される。

実行結果は通常、画面に表示されるだけで、その結果を他のコマンドが再利用することはできない。複数のコマンドを組み合わせて実行したいときはバッチファイルと呼ばれるテキスト形式のファイルを作成するか、パイプと呼ばれる機能を使う。パイプは、複数のコマンドを「|」という記号で区切って1行で打ち込むことで利用できる。

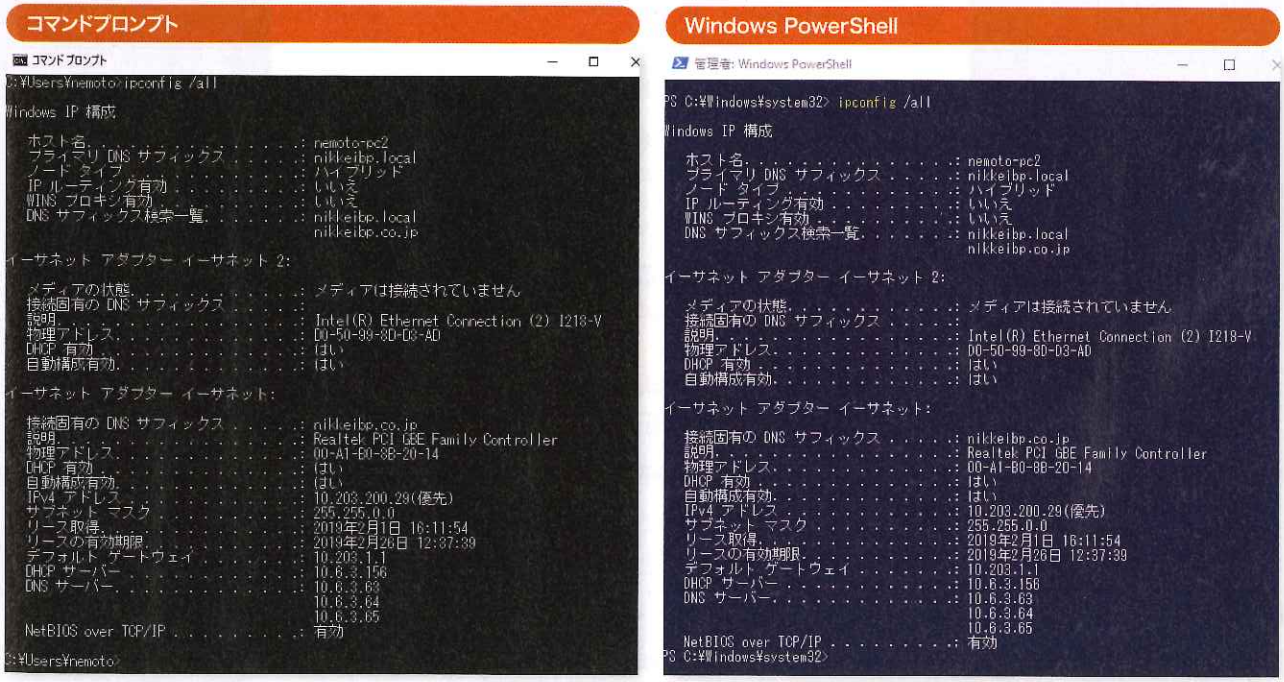
高機能なPowerShell

もう1つのPowerShellはコマンドプロンプトより高機能だ。Windows 7以降のWindowsに標準で搭載されている。

PowerShellは、実行した結果をシェル上で定義した変数として格納できる。単純な数字や文字列だけでなく、Excelのような表形式のデータにも対応する。変数として格納したデータを別のコマンドで加工することができるので、

図1-2 ネットワークコマンドはシェル上で実行する
シェルを使うことで、OSのネットワーク機能を、人間が分かりやすいコマンドとして簡単に利用できる。

図1-3 Windows 10で使える2つのシェル
コマンドプロンプト（左）とWindows PowerShell（右）の2つのシェルが用意されている。同じコマンドを使えるが見た目が若干異なる。PowerShellは高機能だが起動などがやや遅いため、ネットワークコマンドを使うだけならコマンドプロンプトのほうが便利だ。



▼PowerShellを採用している
Windows 10がリリースされた当初は、標準のシェルはコマンドプロンプトだったが、その後PowerShellに変更された。

複数のコマンドを組み合わせた高度な処理も可能だ。

なおコマンドプロンプトとPowerShellでは、実行できるコマンドが異なる。またPowerShellで実行するコマンドは、厳密には「コマンドレット」と呼ぶ。

コマンドプロンプトのコマンドとPowerShellのコマンドレットには似た機能を提供するものが多い。互換性を保つために、PowerShell上でコマンドプロンプトのコマンドも実行できるようになっている。ただし、コマンドプロンプトのコマンドをPowerShellのコマンドレットに置き換えて処理するため、実行結果が異なる場合がある。

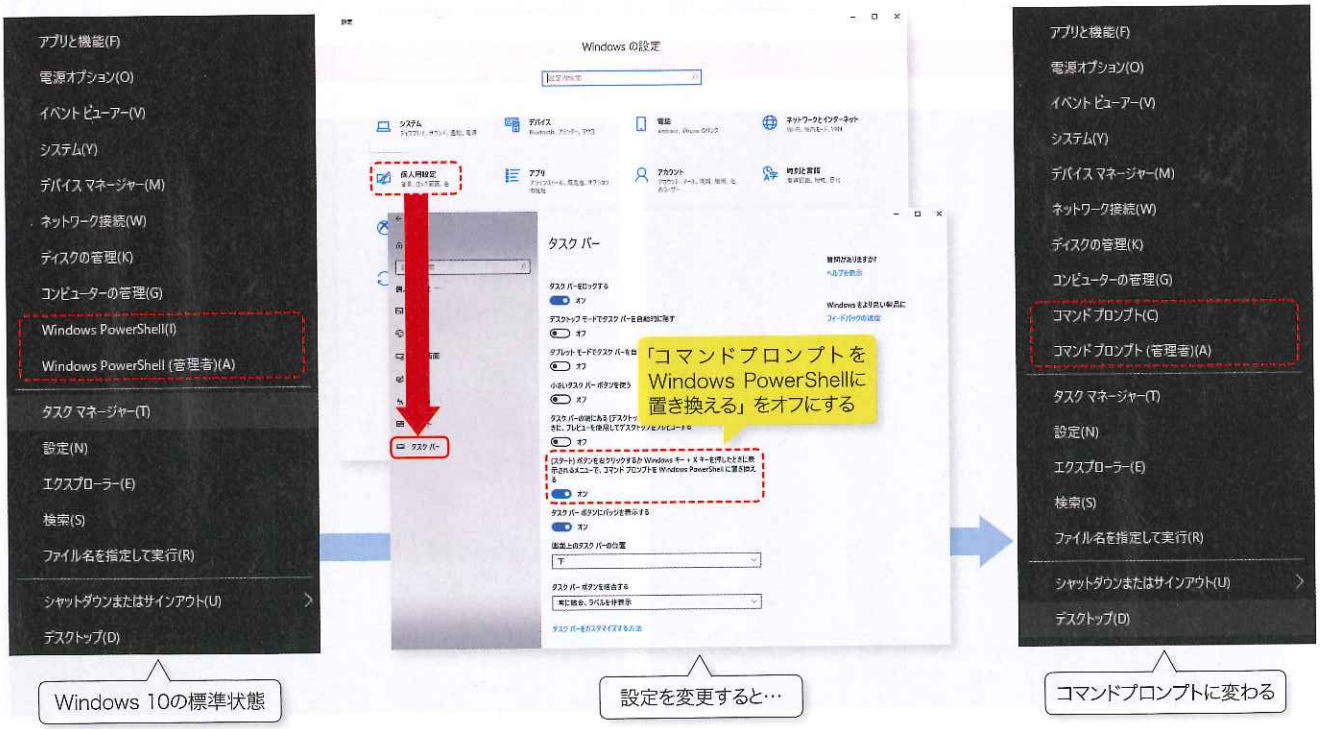
PowerShellが標準のシェルに

Windows 10では、標準のシェルとしてPowerShellを採用している。このため、ネットワークコマンドをコマンドプロンプトで実行しているユーザーは戸惑うことがあるかもしれない。

マイクロソフトとしては、高機能なPowerShellに移行してほしいと考えているようだが、コマンドプロンプトのほうが動作が軽快だ。例えば、コマンドプロンプトの起動に1秒かからないようなコンピューターでも、PowerShellの起動には数秒かかることが珍しくない。前述したように、実行結果が異なる場合もあるため、ネットワークコマンドを使うならコマンドプロンプトで実行したほうがよさそうだ。

Windows 10でも設定を変更すれば、従来と同様にWindowsマークの右クリックからコマンドプロンプトを呼び出せるようにできる（**図1-4**）。設定画面から個人用設定を選び、その中のタスクバーを開くと、「コマンドプロンプトをWindows PowerShellに置き換える」という項目のスイッチがオン

図1-4 デフォルトをコマンドプロンプトにする
Windows 10では初期設定（デフォルト）のシェルがPowerShellになった。タスクバーの設定一覧で「コマンドプロンプトをWindows PowerShellに置き換える」をオフにすると、デフォルトがコマンドプロンプトに変更される。



▼ICMP
Internet Control Message Protocolの略。
▼実装が義務付けられている
ICMPについて規定しているRFC 792に明記されている。

になっているはずだ。ここをオフに変更すれば、右クリックしたメニューからコマンドプロンプトを呼び出せるようになる。

今回の特集では、原則としてコマンドプロンプトを使った場合の画面で、ネットワークコマンドの入力方法や実行結果を紹介する。

ICMPに応答しない

近年、Windowsを取り巻くネットワーク環境が大きく変わっている。それは、ほとんどのWindowsパソコンが、標準でICMP✎に応答しなくなったことだ。ICMPとは、IP通信を手助けするプロトコルである。相手とIPの通信ができるかどうかを確認したり、経路変更やエラーが発生したことを伝えたりするために使う。ネットワークコマンドの多くが、このICMPを使っている。

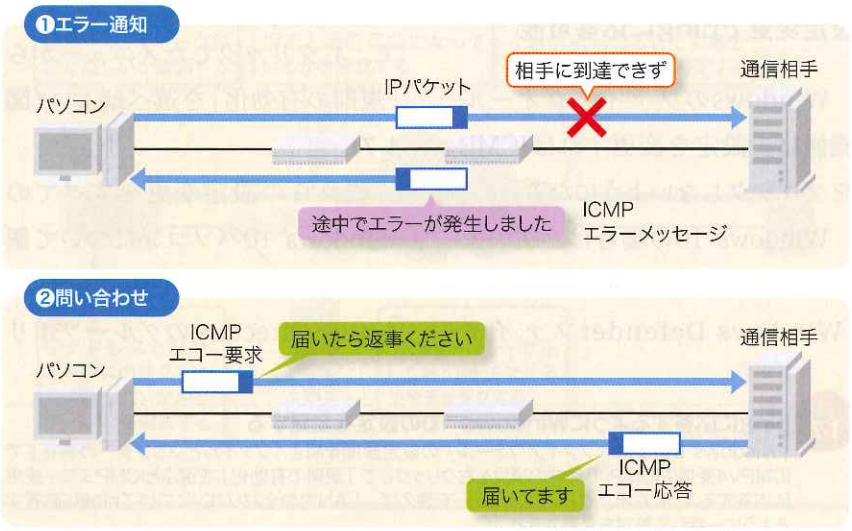
ICMPはIPで通信する機器には実装が義務付けられている✎。ICMPでやりとりするメッセージは「エラー通知」と「問い合わせ」の2種類に大きく分けられる（図1-5）。

エラー通知は経路に障害が発生したり、宛先までの最適経路が変更になったりした場合に、送信元にそれらの情報を通知するために使う。例えば、IPのパケットを受け取ったルーターが転送できなかった場合に、ICMPメッセージでエラーを送信元に伝える。

問い合わせは、相手との通信に問題がないかを調べるものだ。ping（ピング）コマンドでは、ICMPのエコー要求を送ると、それを受け取った相手がエコー応答を返す。これにより、相手と通信できることが確認できる。

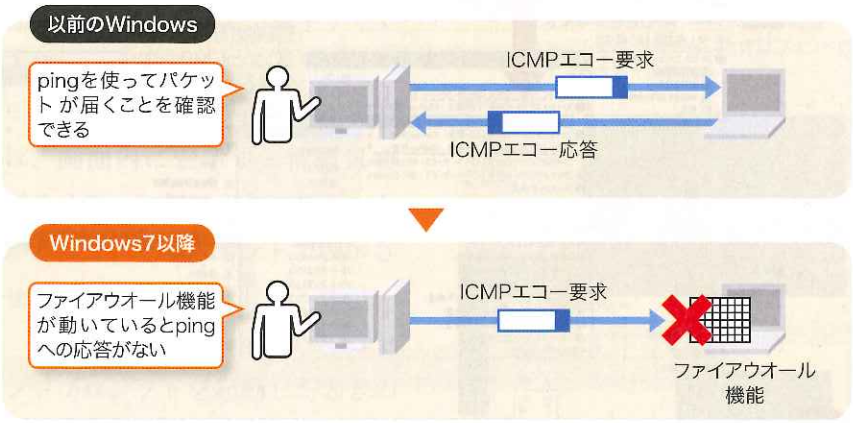
すべてのWindowsがICMPを実装している。しかし、Windowsのファイアウオール機能が有効になっていると、ICMPをブロックしてしまうため応答しないのだ（図1-6）。このため、ネットワークコ

図1-5 **IP通信を手助けするICMP**
ICMPは、IP通信のエラーを通知したり（エラー通知）、宛先の機器と通信できるかを調べたり（問い合わせ）できるプロトコル。IP通信が可能な機器には実装が義務付けられている。いくつかのネットワークコマンドは、ICMPを使ってネットワークの状態を調べる。



ICMP：Internet Control Message Protocol

図1-6 **ファイアウオールがICMPを遮断する**
ICMPを使うpingコマンドで疎通を確認できないなど、従来のトラブル対処法が通用しないことがある。



▼グループポリシー
あらかじめ割り当てられた設定を、Active Directoryにログオンしたときにパソコンあるいはユーザーに対して強制的に適用する機能。

マンドを使ってWindowsパソコンとの疎通や状態を確認することができない。

設定変更でpingに応答可能

Windowsのファイアウオール機能は、設定を変更すればICMPをブロックしないようになる。

Windows 10の場合は、次のように設定を変更する。まず、「Windows Defenderファイアウォール」の設定画面を開く。その中の「受信の規則」にある「ファイルとプリンターの共有(エコー要求)」となっている規則について、右クリックしたメニューから「規則の有効化」を選べばよい(**図1-7**)。

これらの設定変更をすべてのWindows 10パソコンについて個別に実施するのが面倒な場合は、Active Directoryのグループポリシーを使えば、社内のWindowsパソコンに一括して適用できる。IPv4とIPv6のそれぞれに向けた規則があるので、自分の環境に必要なものだけを設定変更すればよい。

このほかにも、Windows 10で変更された点がある。例えば、Windowsのネットワーク機能を操作するnetshコマンドをコマンドプロンプトで実行しようとすると、「Windowsの将来のバージョンで、TCP/IPのNetsh機能が削除される可能性があります」という警告が表示される(**図1-8**)。現状ではコマンドプロンプトでも動作するが、マイクロソフトとしてはPowerShellに移行することを推奨している。

図1-7 pingに応答するようにWindows 10の設定を変更する
「Windows Defenderファイアウォール」の設定画面を開き「ファイルとプリンターの共有」でICMPv4受信やICMPv6受信の項目を右クリックして「規則の有効化」を選ぶとICMPエコー要求に応答するようになる。グループポリシーを使えば、LAN内の全パソコンについてpingに応答するように一括して設定を変更できる。

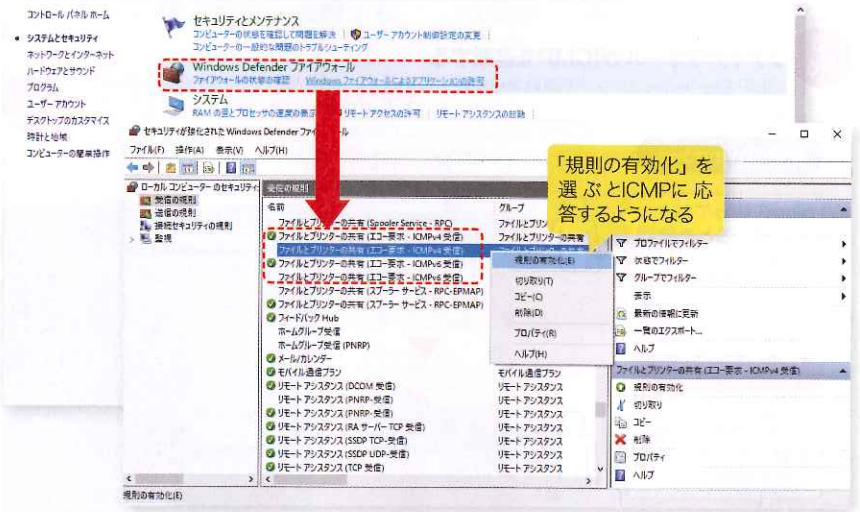
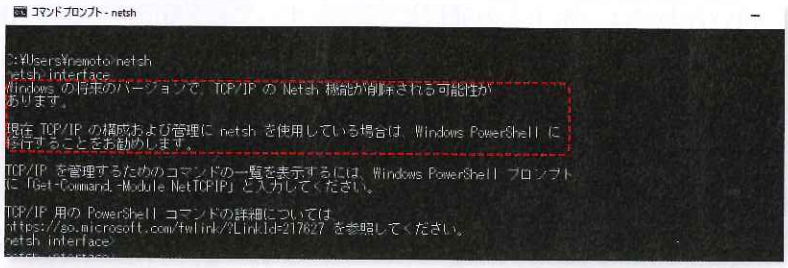


図1-8 netshコマンドを使おうとすると警告が表示される
2019年2月時点ではまだ使えるが、将来的には機能を削除する可能性があると警告し、PowerShellへの移行を推奨している。



Part2では、数多くあるネットワークコマンドの中から、よく使われる5つのコマンドに絞って紹介する(**図1-9**)。これらを押さえれば、ネットワークトラブルの多くに対応できるだろう。

5大コマンドでトラブルに対処

トラブルが発生したとき、まず実行したいのは(1) ipconfig(アイピーコンフィグ)だ。これで、その端末のネットワーク設定に問題がないかを確認するのがトラブル解決の第一歩となる。

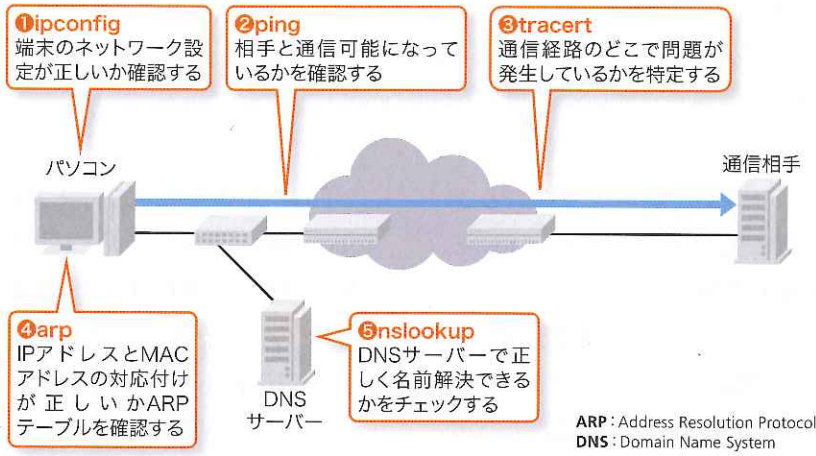
そこで問題が見つからなければ、(2) pingを使って相手と通信できるかどうかを確かめる。次に

▼ARPテーブル
IPアドレスとMACアドレスの対応関係を登録したテーブル。

(3) tracert（トレースルート）で相手までの通信経路を確認する。通信経路の途中で問題が発生していれば、その箇所をある程度特定できる。

それでも解決しなければ、次に(4) arp（アープ）を使って、端末のARPテーブルの情報を確認する。ARPテーブルの情報が古いことがトラブルの原因なら、ARPテーブルを削除することで解決できる。最後に実行するのが(5) nslookup（エヌエスルックアップ）だ。DNSサーバーに問題が発生している場合はこれで分かる。

図1-9 トラブル発生時にネットワークコマンドで調べる基本的な流れ
今回の特集で紹介する5つの基本コマンドを順に使えば、大部分のトラブルについて原因を見つけられる。



ARP：Address Resolution Protocol
DNS：Domain Name System

コマンドプロンプトの文字を大きくする

コマンドプロンプトの画面には一見するとメニューが見当たらない。このため、設定を変更できないように思うかもしれない。だが、左上にあるアイコンの部分をクリックして表示されるプルダウンメニューの中に「プロパティ」という項目がある。

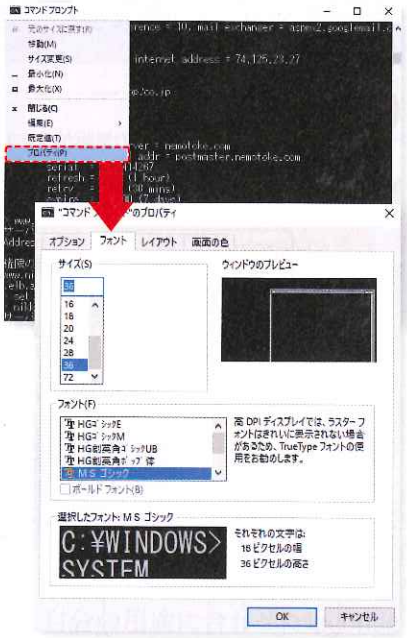
このプロパティのメニューを選ぶと、コマンドプロンプトのいくつかの設定を変更するためのウインドーが表示される（図A）。例えば、「フォント」のタブを選ぶと、コマンドプロンプトの画面表示に使うフォントを変更できる。

コマンドプロンプトの文字が小さくて見づらいと感じている場合は、この画面でサイズを変更すれば文字を大きく変えて見やすくできる。

逆に文字サイズを小さくすれば、画面内に表示する情報量を増やせる。お気に入りのフォントがあれば、そのフォントで表示させることもできる。

このプロパティ画面では、コマンドプロンプトを起動したときの最初のウインドーの大きさや場所、文字や画面の色なども変更できる。コマンドの履歴を保存しておくバッファーサイズの設定もあるので、なるべく多くの履歴を保存しておきたい場合は大きくしておくとよいだろう。

図A コマンドプロンプトのフォントを変える
小さくて文字が見づらいときはフォントのサイズを大きくする。



Part 2 知らなきゃ困る5大コマンド

ipconfig 端末のネットワーク設定を確認

よく使うオプション	
/all	ネットワーク設定情報の詳細を表示する
/renew	ネットワーク設定情報を再設定する
/displaydns	DNSキャッシュの内容を表示する
/flushdns	DNSキャッシュを削除する

パソコンが通信できないとき、最初に実行するのはipconfig（アイピーコンフィグ）だ。実行したWindowsパソコンのIPネットワークの設定情報を確認できる（図2-1）。パソコンが通信できなくなる原因は、設定情報から判明することが多い。

まずは「ipconfig /all」

コマンドプロンプトを開き、オプションなしで「ipconfig」と打ち込んでEnterキーを押すと、IPアドレスやサブネットマスクなどの設定情報が表示される。

詳細な情報を表示するときは、ipconfigの後ろに半角空けて「/all」を付けて実行する（図2-2）。先ほどの実行結果に、ネットワークカード（NIC（ニック））の名前やMAC（マック）アドレス、DHCPサーバー、DNSサーバーが追加される。トラブルシューティングを行うときは、情報量の多いipconfig /allを実行する。

/renewでDHCP情報を再設定

実行結果で最初に確認するのはIPアドレスだ。IPアドレスが表示されない場合は、ネットワークそのものを認識していない。NICの故障やLANケーブルが外れていないかを確認する。

次にIPアドレスそのものを見る。固定IPアドレスを使っているときはそのIPアドレスが表示されているか、DHCPを使っているときはDHCPサーバーが割り当てた設定情報になっているかを確認する。

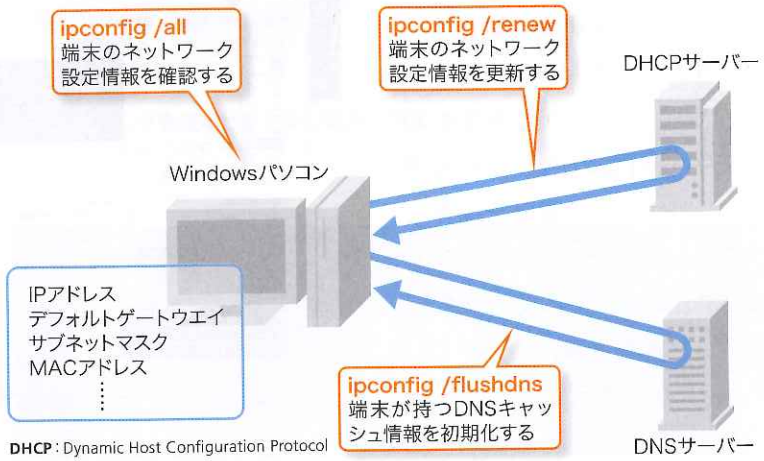
DHCPを使っていて、パソコンの起動時にDHCPサーバーと通信できないと、正しい設定にならない。その場合、「169.254.～」で始まるIPアドレスが設定されることが多い。これは、APIPA（アピパ）と呼ばれる自動設定の機能が割り当てたIPアドレスだ。このIPアドレスでは通信できない。

この場合は、DHCPサーバーにネットワーク設定を改めて割り当てるように要求する。この要求にipconfigを使う。「ipconfig /renew」と実行すると、DHCPサーバーと通信できれば、正しいネットワーク設定に更新される。

ipconfig /renewを実行しても正しいネットワーク設定に更新されない場合は、どこかで障害が発生している。DHCPサーバーの故障やDHCPサーバーとの経路の異常を疑う。

ipconfigを使えば、パソコンが保持しているDNSキャッシュの

図2-1 **ipconfigの基本機能**
IPネットワークを使うための情報がパソコンに正しく設定されているかを確認する。DHCPサーバーやDNSサーバーから受け取った情報を再設定することも可能だ。



▼IPアドレス
ネットワーク上で端末を識別するために使う情報。

▼サブネットマスク
ネットワークの範囲を決める情報。

▼NIC
Network Interface Cardの略。

▼MACアドレス
MACはMedia Access Controlの略。ハードウエアごとに一意に割り当てられる48ビットのアドレス情報。

▼DHCP
Dynamic Host Configuration Protocolの略。ネットワークの設定情報を割り当てに使うプロトコル。

▼DNS
Domain Name Systemの略。

▼APIPA
Automatic Private IP Addressingの略。AutoIPと呼ぶこともある。

▼スクリプト
複数のコマンドをまとめたテキスト形式のファイル。スクリプトを実行するときは、メニューから「コマンドプロンプト（管理者）」を起動して、「powershell -ExecutionPolicy RemoteSigned -File パス付きのスクリプトのファイル名」と打ち込んでEnterキーを押す。

図2-2 ipconfigの実行例

「ipconfig /all」と実行すると、MACアドレスやIPアドレス、サブネットマスクなど、IPネットワークでの通信で使う設定情報が一覧で表示される。これらの情報が、パソコンが装備しているNICごとに表示される。ここでは無線LANを使っている例を示した。有線LANのときは「イーサネット アダプター」などと表示される。

```
C:¥Users¥nemot>ipconfig /all                    ← 入力したコマンド

Windows IP 構成

   ホスト名 . . . . . . . . . . . . : DESKTOP-BF246TI
   プライマリ DNS サフィックス . . . :
   ノード タイプ . . . . . . . . . . : ハイブリッド
   IP ルーティング有効 . . . . . . . : いいえ
   WINS プロキシ有効 . . . . . . . . : いいえ

         ～略～

Wireless LAN adapter Wi-Fi:                     ← ネットワークに接続しているNIC

   接続固有の DNS サフィックス . . . :
   説明 . . . . . . . . . . . . . . : Intel (R) Centrino (R) Advanced-N 6235
   物理アドレス . . . . . . . . . . : B4-B6-76-D6-AE-43   ← MACアドレス
   DHCP 有効 . . . . . . . . . . . : はい
   自動構成有効 . . . . . . . . . . : はい
   IPv6 アドレス . . . . . . . . . . : 2405:6580:34a0:6c00:c81e:f23c:8772:25eb（優先）
   一時 IPv6 アドレス . . . . . . . . : 2405:6580:34a0:6c00:903d:529:234e:26d9（優先）   ← IPv6アドレス
   リンクローカル IPv6 アドレス . . . : fe80::c81e:f23c:8772:25eb%17（優先）
   IPv4 アドレス . . . . . . . . . . : 192.168.179.3（優先）   ← IPv4アドレス
   サブネット マスク . . . . . . . . : 255.255.255.0
   リース取得 . . . . . . . . . . . : 2019年2月6日 10:27:44
   リースの有効期限 . . . . . . . . : 2019年2月7日 10:27:44   ← サブネットマスク
   デフォルト ゲートウェイ . . . . . : 192.168.179.1
   DHCP サーバー . . . . . . . . . . : 192.168.179.1   ← DHCPサーバー
   DHCPv6 IAID . . . . . . . . . . . : 162838134
   DHCPv6 クライアント DUID . . . . : 00-01-00-01-23-CB-72-ED-A4-5D-36-19-8B-EE
   DNS サーバー . . . . . . . . . . : 2405:6580:34a0:6c00:3ae0:8eff:fef1:d1d%17   ← DNSサーバー
                                       192.168.179.1
   NetBIOS over TCP/IP . . . . . . . : 有効
```

DHCPサーバーから受け取った情報

NIC：Network Interface Card

情報を表示したり、削除したりできる。DNSキャッシュは、ドメイン名からIPアドレスを調べるためにDNSサーバーに問い合わせた結果を一時的に保存したものである。この情報が間違っていると、正しい相手と通信できなくなる。内容を確認して必要に応じて削除する。DNSキャッシュの情報の表示には「/displaydns」を、DNSキャッシュの削除には「/flushdns」を付けて実行する。

図2-3 ipconfigの結果を加工するスクリプト

図の内容をファイルとして保存し、コマンドプロンプトからPowerShellを呼び出して実行する（欄外注の「スクリプト」を参照）。

```
$pc1config = ipconfig /all
$pc1config[3].Split(":")[1]
$pc1config[24].Split(":")[1]
$pc1config[27].Split(":")[1]

実行結果
nemoto-pc2
00-A1-B0-8B-20-14
10.203.200.29(優先)
```

「ipconfig /all」の結果をpc1configという変数に格納する

結果の上から数えた行数を指定してコロン（：）以降の抽出を指示する

ホスト名/MACアドレス/IPアドレスの組み合わせを一覧表示

PowerShellで実行結果を加工

PowerShellを使えば、ipconfigの実行結果を変数に格納して、自分に必要な情報だけを取り出すといったことが可能だ。これをスクリプトとして保存しておけば、自分の用途に合わせたipconfigとして利用できる（図2-3）。

ping 相手との通信が可能なことを確認

よく使うオプション	
-n 回数	ICMPエコー要求の送信回数を指定する
-t	ICMPエコー要求を連続して送信する（Ctrl+Cで停止）
-l サイズ	ICMPエコー要求のパケットサイズをバイト単位で指定する
-i TTL	ICMPエコー要求パケットの寿命（TTL）を指定する

pingは、指定した相手とIPネットワークを使った通信が可能かどうか（疎通）を調べるネットワークコマンドである。通信相手の指定には、IPアドレスとホスト名のどちらも使える。

ICMPで疎通を確認

pingは、ICMPを使って相手との疎通を確認する。例えば、「ping IPアドレス」と打ち込んでEnterキーを押すと、IPアドレスで指定した相手に4回のICMPエコー要求を送信する（**図2-4**）。相手はICMPエコー要求を受信すると、ICMPエコー応答を返す。pingはその結果を画面に表示する。

相手と通信できない場合や応答に時間がかかる場合は、「要求がタイムアウトしました」と表示される（**図2-5**上）。問題なく通信できる場合は、応答が返ってくるまでの時間や戻ってきたICMPエコー応答のTTL（TTLについては後述）の値が、4回分それぞれ表示される（同下）。

トラブルの原因を探す

疎通確認だけなら通信相手を指定して実行すればよいが、結果に異常があるときは調べ方を変えてみる。具体的には、ICMPエコー要求を送信する回数を変えたり、ICMPエコー要求のサイズを大きくしたりする。

例えば、4回のうち数回エラーが混ざっていたり、応答時間にばらつきがあったりする場合には、ICMPエコー要求を送信する回数を増やして、どのくらいの頻度でエラーが発生するかを調べる。ICMPエコー要求の回数を指定して実行したいときは「-n 回数」を付けて実行する。ICMPエコー要求を連続して送り続けるときは「-t」を付けて実行する。-tを付けて実行したときは、Ctrlキーを押しながらCキー（Ctrl+C）を押すと、ICMPエコー要求の送信を停止できる。

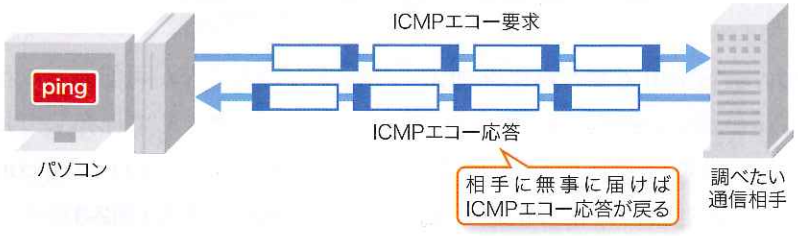
大きなサイズのファイルをやりとりするときだけ時間がかかるといったトラブルが発生した場合は、ICMPエコー要求のサイズを変えて実行する。サイズを徐々に大きくして実行し、応答時間が急激に長くなる前後のサイズを調べる。このような場合、パケットのサイズによって転送に不具合が生じる機器が経路上にあると推測できる。サイズを指定するときは「-l サイズ」（バイト単位で指定）を付けて実行する。

通信相手と疎通できないときは、その経路上にあるネットワーク機器と疎通確認を行う。このとき、最初に指定するのは最も手前にあるデフォルトゲートウエイやルーターだ。デフォルトゲートウエイで障害が発生していたり、ネットワークの設定でデフォルトゲートウエイの指定が間違っていたりして通信できないことがよくある。

通信相手のOSを推測可能

pingの結果に含まれるTTLは、パケットが通過できるルーターの数を示した値だ。ルーターを通

図2-4 pingの仕組み
宛先として指定した相手に対してICMPエコー要求を複数回（デフォルトで4回）送る。それらに対して相手からICMPエコー応答がきちんと戻ってくるかで通信可能かどうかを判断する。



▼TTL
Time To Liveの略。TTLの値はルーターを通過するたびに減っていく。

▼タスクスケジューラ
Windowsが標準で装備している自動実行機能。定期的に実行したいファイルを「タスク」として登録すれば、あらかじめ設定した時間に自動的に実行される。タスクスケジューラはコントロールパネルのシステムツールから呼び出せる。

図2-5 pingの実行例

IPアドレスやホスト名を指定して実行すると、その相手にICMPエコー要求パケットを送信して、その応答を表示する。相手に届かなかった場合は「要求がタイムアウトしました」または「転送中にTTLが期限切れになりました」といったエラーメッセージを、相手に届いた場合は応答時間やTTLが表示される。

```
C:¥Users¥nemoto>ping tech.nikkeibp.co.jp

elb-ex-bpcms-tech-001-1145257536.ap-northeast-1.elb.amazonaws.com [54.92.17.183]に ping を送信しています 32 バイトのデータ:
要求がタイムアウトしました。
要求がタイムアウトしました。
要求がタイムアウトしました。
要求がタイムアウトしました。

54.92.17.183 の ping 統計:
    パケット数: 送信 = 4、受信 = 0、損失 = 4 (100% の損失)、

C:¥Users¥nemoto>ping www.nikkei.com

www-nikkei-com-689290018.ap-northeast-1.elb.amazonaws.com [54.250.186.188]に ping を送信しています 32 バイトのデータ:
54.250.186.188 からの応答: バイト数 =32 時間 =15ms TTL=235
54.250.186.188 からの応答: バイト数 =32 時間 =9ms TTL=235
54.250.186.188 からの応答: バイト数 =32 時間 =9ms TTL=235
54.250.186.188 からの応答: バイト数 =32 時間 =9ms TTL=235

54.250.186.188 の ping 統計:
    パケット数: 送信 = 4、受信 = 4、損失 = 0 (0% の損失)、
ラウンド トリップの概算時間 (ミリ秒):
    最小 = 9ms、最大 = 15ms、平均 = 10ms
```

入力したコマンド
入力したコマンド
相手に届かなかった場合のメッセージ
TTLの値で相手のOSが推測できる
相手に届いた場合のメッセージ
応答が返ってくるまでの時間

過するごとに1ずつ小さくなり、0になったパケットは廃棄される。

TTLの初期値は、パケットを送信する機器が決める。一般的にはOSによって異なり、Windowsが動く機器は128、macOSやLinuxは64、Solarisは255である。ネットワーク機器は、製品によって異なる。

ICMPエコー応答が届くまでに経由するルーターが数十台に及ぶことはまずない。通常は初期値より少し減った値が表示されるので、その値から通信相手の機器のOSを推測できる。

定期実行して通常状態を把握

PowerShellを使えば、pingと同様の処理を定期的に実行し、その結果を残しておくことができる。例えば、**図2-6**のようなスクリプトを作成して、タスクスケジューラに登録しておけば、応答時間の平均、最小、最大の値を定期的に記録できる。

応答時間を定期的に記録しておけば、トラブルシューティングのときに役立つ。「いつもより通信に時間がかかる」といったトラブルが発生したとき、通常時の応答時間との比較や、いつごろから変化が見られるのかといったことが分かる。

図2-6 PowerShellを使ってpingと同じ処理を実行する

スクリプトを実行すると、その日時と応答時間の平均/最小/最大が記録される。この例のように日時によって反応がなかったり遅かったりした場合は、通信が不安定だと推測できる。

```
$cmd="ping example.com"
$out="d:¥temp¥ping-result.txt"
$exdate=date
$exdate=$exdate -replace " "
$result=Invoke-Expression $cmd
if ($result.Count -eq 11) {
 $tmp=$result[10]
 $tmp2 = $tmp -replace "ms、",","
 $tmp2  = $tmp2 -replace "="
 $tmp2  = $tmp2 -replace "平均"
 $tmp2  = $tmp2 -replace "最小"
 $tmp2  = $tmp2 -replace "最大"
 $tmp2  = $tmp2 -replace "ms"
 $tmp2  = $tmp2 -replace " "
} else {
 $tmp2="-,-,-"
}
$result=$exdate+","+$tmp2
echo $result >> $out
```

ping-result.txt - メモ帳
ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)

```
2019/02/0517:54:03,9,9,9
2019/02/0517:57:47,5,8,5
2019/02/0518:19:59,-,-,-
2019/02/0519:38:02,117,117,117
```

実行日時
応答時間の平均、最小、最大

tracert 相手との通信が通過する経路を調査

よく使うオプション

-d	IPアドレスの逆引きをしない
-w 時間	タイムアウトの時間をミリ秒単位で指定（デフォルトは4秒）
-h ルーター数	調べる最大のルーター数を指定する（デフォルトは30）

tracert（トレースルート）は、通信相手までの経路を調査するネットワークコマンドだ。相手と通信できないときに、pingでデフォルトゲートウエイまでの疎通確認ができたら、tracertを使って障害が発生している箇所を探し出す。

エラーメッセージを利用する

tracertは、ICMPを利用して経路上にあるルーターを一覧表示する。最初にTTLの値を1に設定したICMPエコー要求パケットを送り、TTLが0になったときにルーターが送信元に送る「転送中にTTLが期限切れになりました」というエラーメッセージが届くのを待つ（図2-7）。戻ってきたら、次にTTLの値を2に設定したICMPエコー要求パケットを送り、応答を待つ。TTLの値を順に増やしながら、通信相手が応答を返すまでこれを繰り返す。こうして、経路上のルーターの状態を調べる。

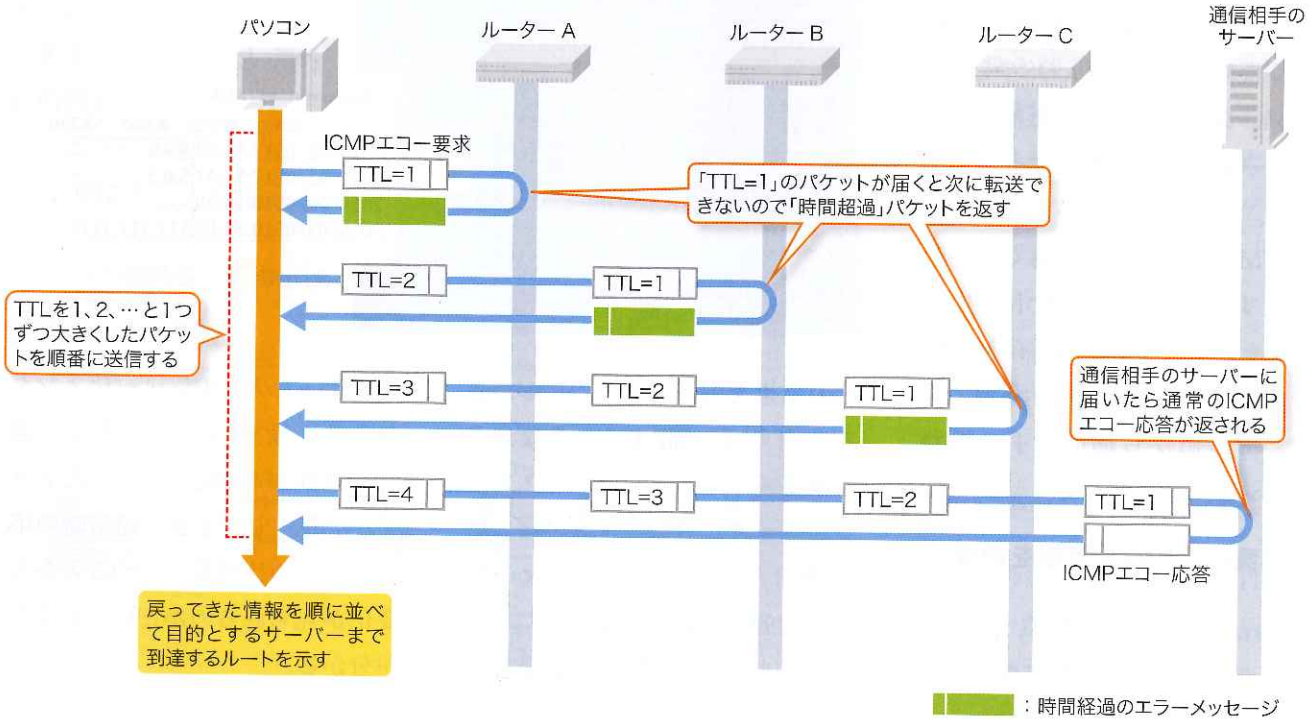
逆引きをやめて時間短縮

「tracert IPアドレスまたはホスト名」を実行すると、経路上にあるルーターが縦に一覧表示され、それぞれのルーターからエラーメッセージが届くまでにかかった応答時間が表示される（図2-8）。

ルーターは、IPアドレスまたはホスト名で示される。ネットワーク機器は通常ホスト名を設定しないため、IPアドレスで表示されることが多い。にもかかわらず、tracertはデフォルトで、IPアドレスをDNSサーバーで逆引きを行ってホスト名を表示しようとする。このためtracertでは、逆引きに失敗（タイムアウト）するまで何度も待たされることになる。トラブルシューティングなど、迅速に調査する必要があるときは、IPアドレスの逆引きをしないように「-d」を付けて実行する。

図2-7 **tracertの仕組み**

TTLを1から順に増やしながら送信することで、目的のサーバーまでの間にあるルーターなどの情報を調べる。TTLは、パケット転送の最大回数を示す。



▼TTLの値を1に設定したICMPエコー要求パケット
pingでも同様のことを実行できる。「ping -i 1 通信相手」と実行する。

図2-8 tracertの実行例
TTLを1ずつ増やしながら実行した結果を順に並べることで、目的のサーバーまでの経路が分かる。最近ではセキュリティー対策などで、ICMPに応答しないルーターやICMPパケットを通過させない機器があるため、途中でタイムアウトと表示されることが多い。

```
C:¥Users¥nemoto>tracert www.nikkei.com

www-nikkei-com-689290018.ap-northeast-1.elb.amazonaws.com [54.250.186.188] へのルートをトレースして います
経由するホップ数は最大 30 です:

  1   <1 ms    <1 ms    <1 ms  10.203.1.2
  2    1 ms     1 ms     3 ms  10.252.1.4
  3    1 ms     1 ms     1 ms  10.3.1.10
  4    2 ms     2 ms     2 ms  cat35608.nikkeibp.co.jp [101.110.52.145]
  5   15 ms     4 ms     4 ms  softbank126112253017.biz.bbtec.net [126.112.253.17]
  6   10 ms     5 ms     5 ms  10.0.192.33
  7    9 ms     8 ms     5 ms  10.0.60.105
  8    5 ms     5 ms     5 ms  10.9.203.206
  9    6 ms     5 ms     6 ms  softbank221110131174.bbtec.net [221.110.131.174]
 10    *        *        *     要求がタイムアウトしました。
 11    *        *        *     要求がタイムアウトしました。
 12    *        *        *     要求がタイムアウトしました。
 13    *        *        *     要求がタイムアウトしました。
 14    *        *        *     要求がタイムアウトしました。
 15    *        *        *     要求がタイムアウトしました。
 16    *        *        *     要求がタイムアウトしました。
 17    *        *        *     要求がタイムアウトしました。
 18    *        *        *     要求がタイムアウトしました。
 19    8 ms     8 ms    10 ms  52.95.31.13
 20    6 ms     6 ms     7 ms  52.95.31.211
 21    8 ms    11 ms     9 ms  52.95.31.168
 22   10 ms    10 ms    10 ms  52.95.31.130
 23    9 ms     9 ms     9 ms  52.95.30.212
 24    *        *        *     要求がタイムアウトしました。
 25    *        *        *     要求がタイムアウトしました。
 26    *        *        *     要求がタイムアウトしました。
 27    *        *        *     要求がタイムアウトしました。
 28    *        *        *     要求がタイムアウトしました。
 29    9 ms     9 ms     9 ms  ec2-54-250-186-188.ap-northeast-1.compute.amazonaws.com [54.250.186.188]

トレースを完了しました。
```

入力したコマンド
目的のサーバーまでのルーターが順に並ぶ
TTLを1ずつ増やしながら順に実行する
4秒以内に応答がないとタイムアウトになる
実行した3回の応答時間

セキュリティー対策などで、ICMPの応答を返さないルーターがあったり、経路上にICMPパケットを通過させない機器があったりすると、tracertの実行に時間がかかる。通過させない機器が経路上にあるときは、その機器より先の結果はすべて「要求がタイムアウトしました」になる。より迅速に結果を知りたいときは、タイムアウトの時間とルーター数の上限を指定するとよい。「-w」でタイムアウトの時間を、「-h」でルーター数の上限を指定できる。

過去の履歴を呼び出して再利用

ネットワークコマンドを使っていて、以前実行したコマンドを再度実行したいことはよくある。こうしたときに便利なのが履歴の呼び出しだ。コマンドプロンプトの画面でF7キーを押せば、過去に実行したコマンドを呼び出して再利用することが可能だ（図B）。全く同じままではなく、一部を編集して実行することも可能だ。

図B コマンドプロンプトで履歴を呼び出す
F7キーを押すと、これまでに実行した過去のコマンドを呼び出して、再度実行できる。

arp IPアドレスとMACアドレスの対応付けを管理

よく使うオプション

-d	機器のARPテーブルの内容を削除
-d IPアドレス	機器のARPテーブルから指定したIPアドレスの情報を削除
-a	機器のARPテーブルを表示

arp（アープ）はARPテーブルを管理するためのコマンドだ。ARPテーブルとは、同じネットワーク内にあるパソコンなどの機器のIPアドレスとMACアドレスの対応付けを一時的に保持しておくためのものである。

機器を入れ替えたり、同じIPアドレスを引き継ぐ別の機器が稼働したりすると、ARPテーブルを更新する必要がある。こういったときにarpを使えば、ARPテーブルの内容を表示したり、削除したりできる。

MACアドレスを宛先にする

IPネットワークで通信する際には、通信相手をIPアドレスで指定する。しかし、IPパケットを運ぶイーサネットフレームでは、フレームを次に渡す相手をMACアドレスで指定する。このためIPネットワークであっても、MACアドレスが分からないと通信できない。

IPアドレスで宛先を指定されたパケットは、宛先が同じネットワークにあれば通信相手に直接送信される。宛先が別のネットワークにあれば、デフォルトゲートウエイに送られる。

このときイーサネットフレームを使うため、通信相手またはデフォルトゲートウエイのMACアドレスを調べる必要がある。具体的には、IPアドレスを記述したARP要求パケットを同じネットワークの全ての機器に向けて送信し、該当する機器から応答が届くのを待つ（**図2-9**）。応答が届けば、応答の送信元MACアドレスを宛先にしたイーサネットフレームを送る。

この作業を毎回実行するのは非効率なので、ARPテーブルを保持するようにしている。もしARPテーブルに載っている機器にイーサネットフレームを送るときは、ARP要求パケットを送らずに、ARPテーブルを参照する。こうすることで、すぐにイーサネットフレームを送信できて、ネットワークの混雑も緩和できる。

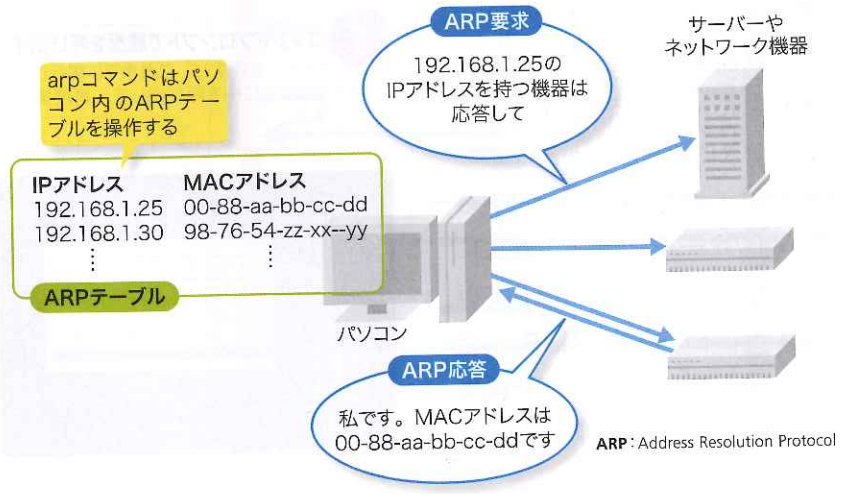
ARPテーブルの内容を消す

このARPテーブルの仕組みが、逆にトラブルを引き起こすことがある。何らかの原因でARPテーブルで保持している情報と、実際のネットワークでの情報が食い違ってしまった場合だ。

例えば、障害が発生したときにバックアップ機が立ち上がり、同じIPアドレスで通信を始めたときなどが典型的だろう。機器が異なるのでMACアドレスも異なるが、IPアドレスは以前の機器と同じだ。こうなると、LAN上に存在しない相手にデータを送ることになり通信に失敗する。

通信エラーが発生するときは、arpコマンドを使ってARPテーブ

図2-9 ARPの基本機能
コマンドを実行するパソコンで保持しているARPテーブルを操作する。ARPテーブルはARP要求に応えたARP応答の内容を一時的にキャッシュしておくものだ。



▼ARP
Address Resolution Protocolの略。
▼CSV
Comma Separated Valuesの略。

ルを削除するだけで解消できることがある。ただし、ARPテーブルを削除するには、管理者権限が必要になる。具体的には、メニューの「コマンドプロンプト（管理者）」でコマンドプロンプトを起動してから、「arp -d」を実行して、その機器のARPテーブルの内容を削除する。

トラブルが発生している機器が分かっているときは、「-d IPアドレス」を付けて実行すれば、該当するIPアドレスとそのMACアドレスだけを削除できる。ただARPテーブルの情報はもともと長時間は保持しない。トラブルが発生したときは、IPアドレスを指定せずに、まとめて削除したほうが手間が少なくてよいだろう。

ARPテーブルの内容を表示する

「-a」を付けて実行すると、その時点でARPテーブルに保持されている内容を確認できる（図2-10）。この中の「インターネットアドレス」がIPアドレス、「物理アドレス」がMACアドレスに該当する。

右にある「種類」は、そのARPの情報が登録された状況を示す。「動的」となっているのが、通常のARP要求とAPR応答のやりとりで取得した対応付けを意味する。一方「静的」となっているものは、DHCPサーバーから受け取った内容をもとにOSが登録した情報である。

図2-10 arpの実行例
-aオプションを付けて実行すると、機器に保持されているARPテーブルの情報を一覧表示する。

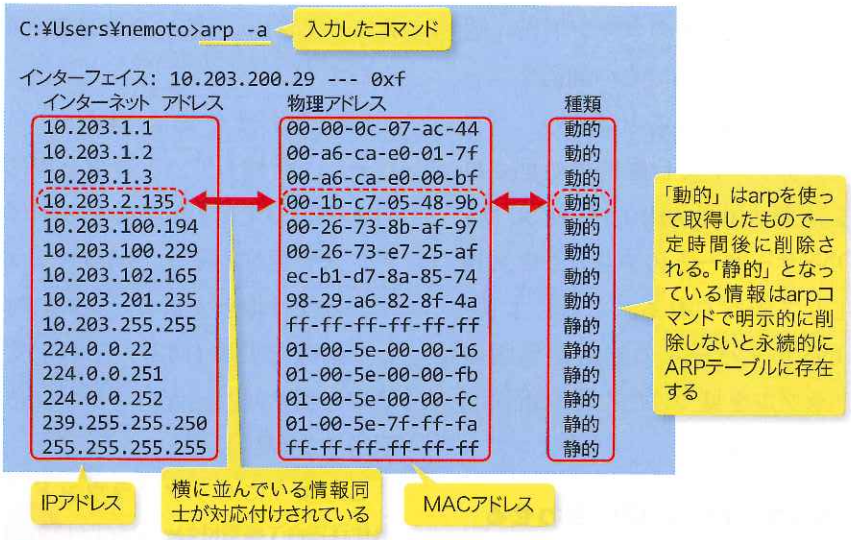
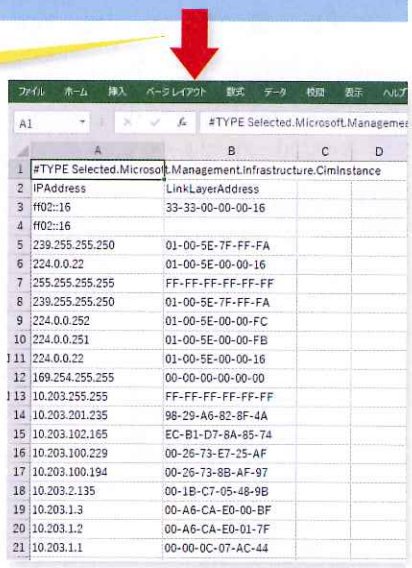


図2-11 PowerShellを使うとARPテーブルの情報を自由に加工できる
CSV形式のデータとしてファイルに保存したり、必要な情報だけを取り出したりすることが可能だ。

```
Get-NetNeighbor | Select-Object IPAddress,LinkLayerAddress |
Export-Csv -Path .¥arp.csv
```

ARPテーブルの内容をCSV形式のファイルとして保存

CSV：Comma Separated Values



	A	B	C	D
1	#TYPE Selected.Microsoft.Management.Infrastructure.CimInstance			
2	IPAddress	LinkLayerAddress		
3	ff02::16	33-33-00-00-00-16		
4	ff02::16			
5	239.255.255.250	01-00-5E-7F-FF-FA		
6	224.0.0.22	01-00-5E-00-00-16		
7	255.255.255.255	FF-FF-FF-FF-FF-FF		
8	239.255.255.250	01-00-5E-7F-FF-FA		
9	224.0.0.252	01-00-5E-00-00-FC		
10	224.0.0.251	01-00-5E-00-00-FB		
11	224.0.0.22	01-00-5E-00-00-16		
12	169.254.255.255	00-00-00-00-00-00		
13	10.203.255.255	FF-FF-FF-FF-FF-FF		
14	10.203.201.235	98-29-A6-82-8F-4A		
15	10.203.102.165	EC-B1-D7-8A-85-74		
16	10.203.100.229	00-26-73-E7-25-AF		
17	10.203.100.194	00-26-73-8B-AF-97		
18	10.203.2.135	00-1B-C7-05-48-9B		
19	10.203.1.3	00-A6-CA-E0-00-BF		
20	10.203.1.2	00-A6-CA-E0-01-7F		
21	10.203.1.1	00-00-0C-07-AC-44		

動的の情報は一定時間後に削除されるが、静的となっている情報はarpコマンドで明示的に削除しない限りARPテーブルに永続的に存在する。

CSV形式で出力する

図2-11に示したのは、PowerShellを使ってARPテーブルの内容をCSVファイルに出力するスクリプトだ。実行すると、右下に示したCSV形式のデータを作成できる。ここでは、IPアドレスとMACアドレスを抜き出すスクリプトを示した。

nslookup DNSを使った名前解決を確認

よく使うオプション

server サーバー名	名前解決を問い合わせるDNSサーバーを指定
set 環境変数=値	環境変数の値を指定

nslookup（エヌエスルックアップ）はDNSサーバーとやりとりするためのコマンドだ。通常はDNSサーバーとの通信をユーザーが意識することはない。バックグラウンドで自動的に処理される。だがnslookupを使えば、DNSサーバーとのやりとりを可視化して調べられる。これにより、DNSサーバーとの通信が原因のトラブルを見つけ出したり解決したりできる。

DNSサーバーに問い合わせる

例えば、Webブラウザーでwww.nikkeibp.co.jpにアクセスする場合を考えてみよう（**図2-12**）。

アドレス欄に「www.nikkeibp.co.jp」と入力すると、WebブラウザーはまずDNSサーバーに「www.nikkeibp.co.jpのIPアドレスを教えてください」と問い合わせる（図2-12の1）。

問い合わせを受けたDNSサーバーは、管理している情報を基に「10.6.3.63です」と該当するIPアドレスを返す（同2）。Webブラウザーは受け取ったIPアドレスに対して情報を要求（同3）。それを受け取ったサーバーがコンテンツを転送する（同4）。

このようにバックグラウンドで実施しているDNSサーバーとのやりとりを、nslookupを使えば1つ1つ試せる。

もしDNSサーバーの設定が間違っていたり、一時的にダウンしていたりすると、問い合わせに対する返答が戻ってこないのですぐに分かる。

正引きと逆引きが可能

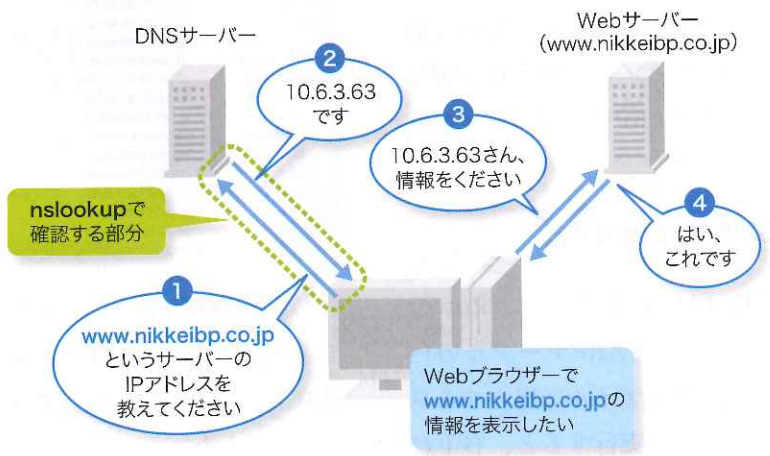
DNSサーバーに問い合わせることを「名前解決」と呼び、ドメイン名からIPアドレスを調べることを「正引き」、IPアドレスからドメイン名を調べることを「逆引き」と呼ぶ。nslookupの後ろにドメイン名を指定すれば正引き、IPアドレスを指定すれば逆引きを実行する。

実際に正引きを実行した結果が**図2-13**だ。ここでは、ドメイン名を指定して正引きを試している。その結果として、問い合わせ先のDNSサーバーのホスト名とIPアドレスに続いて、問い合わせへの回答となるIPアドレスの情報が返される。このドメインでは2つのIPアドレスがDNSサーバーに登録されていることが分かる。

nslookupコマンドでは、問い合わせるDNSサーバーを指定することも可能だ。実際に「8.8.8.8」というDNSサーバーを指定して、同じドメイン名への正引きを実行した結果が図2-13の下側だ。問い合わせ先は変わったが、同じIPアドレスが返されている。これにより、DNSサーバーには問題が発生していないと推測できる。

なお、DNSサーバーを指定した2回目の問い合わせの結果には、「権限のない回答」と表示されてい

図2-12 nslookupの役割
ホスト名を使ってサーバーにアクセスするには、ホスト名から該当するサーバーのIPアドレスを調べる名前解決が必要だ。nslookupを使えば、DNSサーバーへの問い合わせの部分で問題が発生していないかを確認できる。



▼MXレコード
MXはMail eXchangeの略。DNSサーバーに登録する情報の1つで、そのドメイン宛てのメールを転送するアドレスを記述する。

る。これは、問い合わせたDNSサーバーがキャッシュDNSサーバーであることを示している。

DNSサーバーには、権威DNSサーバーとキャッシュDNSサーバーの2種類がある。権威DNSサーバーは、組織のIPアドレス情報などを管理するサーバーを指す。世界中のDNSサーバーと連携して分散データベース環境を構成している。これに対し、権威DNSサーバーに問い合わせた結果を一時保存し、問い合わせに対して代わりに答えるサーバーをキャッシュDNSサーバーと呼ぶ。

対話型でも操作できる

nslookupは、コマンドプロンプトから1行で実行するだけでなく、対話型でも実行できる。例えばドメイン名やIPアドレスを指定せずに「nslookup」とだけ入力すると、プロンプトの形が変化し、nslookupの対話モードに切り替わる(図2-14)。

nslookupの対話モードでは、コマンドプロンプトのときと操作方法が異なる。例えば、問い合わせの対象とするレコードをMXレコードに切り替えてから問い合わせを実行、といったように段階を踏んで行う。

対話モードを終了するときは「exit」と入力する。そうすると、通常のコマンドプロンプトに戻る。

図2-13 nslookupの実行例
サーバー名を指定すれば、IPアドレス情報が返される。標準では、ipconfig /allで表示されるDNSサーバーに問い合わせるが、問い合わせ先のDNSサーバーを明示的に指定することも可能だ。

```
C:¥Users¥nemoto>nslookup www.nikkeibp.co.jp
サーバー:  bpdc1.nikkeibp.co.jp
Address:  10.6.3.63

名前:    elb-ex-bpcms-common-001-1985407959.ap-northeast-1.elb.amazonaws.com
Addresses:  54.249.217.37
          54.65.7.40
Aliases:  www.nikkeibp.co.jp

C:¥Users¥nemoto>nslookup www.nikkeibp.co.jp 8.8.8.8
サーバー:  google-public-dns-a.google.com
Address:  8.8.8.8

権限のない回答:
名前:    www.nikkeibp.co.jp
Addresses:  54.65.7.40
          54.249.217.37
```

入力したコマンド
問い合わせ先DNSサーバーの情報。標準ではipconfigで表示されるDNSサーバーに問い合わせる
DNSサーバーが返したIPアドレス
入力したコマンド
問い合わせ先のDNSサーバーを明示的に指定することも可能
権威DNSサーバーではなくキャッシュDNSサーバーに問い合わせた場合に表示される

図2-14 nslookupは対話型でも実行できる
問い合わせる情報のタイプを指定すれば、メールサーバー名を格納するMXレコードや、ドメイン情報を格納するSOAレコードなどについても調べられる。

```
C:¥Users¥nemoto>nslookup
既定のサーバー:  bpdc1.nikkeibp.co.jp
Address:  10.6.3.63

> set type=mx
> ganchans.net
サーバー:  bpdc1.nikkeibp.co.jp
Address:  10.6.3.63

権限のない回答:
ganchans.net    MX preference = 5, mail exchanger = alt1.aspmx.l.google.com
ganchans.net    MX preference = 10, mail exchanger = aspmx3.googlemail.com
ganchans.net    MX preference = 10, mail exchanger = aspmx5.googlemail.com
ganchans.net    MX preference = 1, mail exchanger = aspmx.l.google.com
ganchans.net    MX preference = 5, mail exchanger = alt2.aspmx.l.google.com
ganchans.net    MX preference = 10, mail exchanger = aspmx4.googlemail.com
ganchans.net    MX preference = 10, mail exchanger = aspmx2.googlemail.com

aspmx.l.google.com      internet address = 74.125.23.27
> set type=soa
> nemotoke.com
サーバー:  bpdc1.nikkeibp.co.jp
Address:  10.6.3.63

権限のない回答:
nemotoke.com
        primary name server = nemotoke.com
        responsible mail addr = postmaster.nemotoke.com
        serial  = 1309414267
        refresh = 3600 (1 hour)
        retry   = 1800 (30 mins)
        expire  = 604800 (7 days)
        default TTL = 3600 (1 hour)
```

プロンプトの表示が変わる
単純に「nslookup」とだけ入力すると対話モードになる
MXレコードを指定
メールサーバーのホスト名
SOAレコードを指定
登録されているドメイン情報