

C-Native

クラウドシフトへの第一歩は、
「C-Native」から

伴走型支援

パッケージプラン

短期導入



C-Native Transformation Service



@IT > クラウド > Windows Server Insider > 第14回 信頼性のある通信を実現するTCPプロトコル...

第14回 信頼性のある通信を実現するTCPプロトコル（その1）

(2/3 ページ)

2003年12月25日 00時00分 公開

[デジタルアドバンテージ, 著]

印刷

通知

見る

Share

15

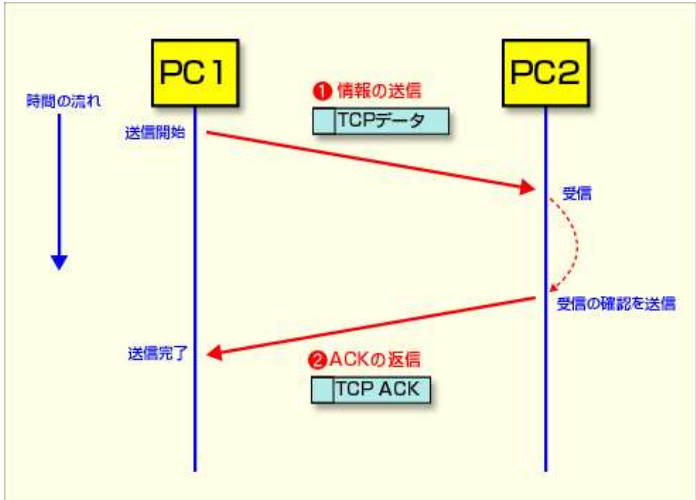
前のページへ

123

次のページへ

TCPでは、信頼性のある通信を実現するために、相手にデータが届いたことを必ず確認するという手法を採用している。データを送信する場合、それに対する確認応答を受け取って始めて送信が完了したと判断するのである。

以下は、PC1からPC2に対して、TCPを使ってデータを送信する場合の通信のやりとりを示したものである。



TCPにおける基本的な通信形態

TCPでは、送信したパケットに対して必ず確認のための応答を送信することによって信頼性のある通信を実現している。

(1) 送信するデータをTCPパケットに載せ（送信したデータにTCPヘッダを付加する）、それをIPパケットに載せてPC2あてに送信する。

(2) TCPパケットを受け取ると、データを受け取ったことを示すためにACK/パケット（正確には、ACKフラグがオンになったTCPパケット）を送信する。ACKを受信して始めて、送信が正常に終了したことになる。

まずPC1が送信したいデータをTCPのパケットに組み込んで、PC2に向けて送信する（図中の **(1)**）。もちろん実際には、送信したいデータにTCPヘッダを付加し、IPパケットに載せて相手のマシンにまで送信するのである。IPは信頼性のない、データグラム通信を実現するためのプロトコルであることはすでに連載10回「IPパケットの構造とIPフラグメンテーション」で述べた通りである。そのため、ネットワークの状況などによっては相手まで届かないこともあるが、ほとんどの場合はこれによって相手にまでTCPパケットが届けられることになる。

TCPパケットを受信した側では、TCPパケットからデータを取り出して上位アプリケーションに届けるだけでなく、送信側に対して自動的に確認応答を返信する。確認応答の返信は、上位アプリケーションの指示とは関係なく、TCPのプロトコル・スタック内で自動的に行われる。このような仕組みになっているため、上位アプリケーションは、TCPプロトコルにおけるパケットのやりとりをまったく意識することなく、信頼性のあるストリーム通信機能を利用することができる。

検索

ホワイトペーパー

- 

次世代ネットワーク実現の鍵、スイッチングインフラの最新要件とは？
- 

障害対応を迅速化、ネットワーク監視ツールの選定で押さえるべき3つのポイント
- 

ネットワーク製品の導入に関する読者調査レポート(2014年12月)
- 

もう「Wi-Fi 7」時代？ 無線LANの気になる進化



クラウドシフトへの第一歩は、「C-Native」から

C-Native Transformation Service

伴走型支援

パッケージプラン

短期導入

C-Native の新サービスを詳しく

スポンサーからのお知らせ

中堅中小企業の“ネットワーク課題”はこれで解決！

重要なのは発展性 なぜ今、“ストレージ”に注目が集まっているのか

Special

- 

複数ベンダーの「継ぎはぎSASE」で生じる課題、どうすれば解決できるのか？
- 

中堅中小企業の“ネットワーク課題”はこれで解決！ **New!**
- 

データは「守りながら活用する時代」に
- 

「守る」だけでは不十分 今どきのストレージには何が必要？
- 

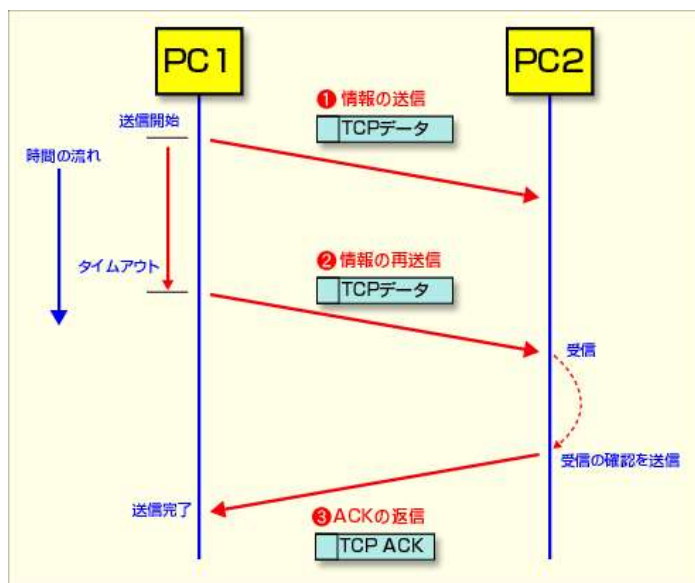
ローコードツールの現在地。AI、機械学習とのシナジーで新たな価値を生み出す **New!**
- 

社内ルールだけでは限界 有名無実化した「ローカル保存禁止」にどう対応？
- 

オンプレのITインフラを「サブスク」で利用できるサービスは何がスゴイのか？

確認応答のことをTCPでは「Acknowledge（承認）」と呼ぶ（以下ではACKと略記する）。TCPのパケットを受信した側では、データを受け取ったことを表すため、ACK応答を返送する（図中の（2））。

送信側では、このACKを受け取って初めて送信が完了したとみなし、次の動作に移る。だがしばらく待ってもACKが受信できなければ、送信が失敗したものとみなし、次のように再送動作を行う。



TCPパケットの再送信

TCPパケットを送信後、しばらく待ってもACKが受信できなければ、TCPパケットの再送信動作を行う。

- (1) 最初のTCPパケットの送信。
- (2) 一定時間待ってもACKパケットが戻ってこない場合、先ほどと同じTCPパケットを再送信する。再送信するパケットは最初のもと同じ。
- (3) ACKパケットを受け取れば送信完了。これが実際に（1）のパケットに対する応答なのか、それとも（2）のパケットに対する応答なのかは関係ない（どちらのパケットも同じなので、どちらに回答してもよい）。

この例では、1回再送信した後、ACKを受信している。再送信するTCPパケットの内容は、（通常は）最初に送信したTCPパケットの内容とまったく同じである（再送であることを示す特別な情報は含まれていない）。ACKを受信するまで、何度かTCPパケットを再送するが、このような再送動作に対しても、上位アプリケーションは何ら指示・関与する必要はない。すべてTCP/IPのプロトコル・スタック内で自動的に処理される。もしあらかじめ規定された回数（もしくは時間）経ってもACKを受信できない場合は、TCPコネクションが切れてしまったと判断し（通信相手がダウンしてしまったような場合も含む）、上位アプリケーションに対してエラーなどを通知することになる。



Special

- PR -

「守る」だけでは不十分 今どきのストレージには何が必要？

ACKが受信できない要因としてはいろいろなものが考えられる。送信したはずのTCPパケットが、（ネットワーク混雑などの理由により）ルーティングの途中で消失してしまうだけでなく、ACKパケットが消失してしまうという可能性もある。

ACKが受信できない場合、再送処理を最大で何回まで行うか、送信が失敗したと判断する「タイムアウト」時間をいくらにするか（どの程度の時間待ってから再送信を行うか）などのパラメータは、すべてTCP/IPの実装に依存しており、プロトコルとして決められているわけではない。実際のTCP/IPの実装では、タイムアウト時間は固定値ではなく、ネットワークの状況などに応じて動的に変化するように設計されている*1



NTTデータと日本IBMがタッグ！
AIは仕事をどう変える？



「ほとんど誰も見ていない」社内ポータル、どう変えるべき？
New!

[@IT Specialへ](#)

Windows Server Insider 記事ランキング

本日

月間

Excel（エクセル）で日付から自動的に曜日を入力する

【Excel】重複データを色付けして瞬時にダブりをチェックする

【Excel】パスワードロックを強制的に解除する方法

TCP/IP通信の状態を調べる「netstat」コマンドを使いこなす【Windows OS】

Windows OSのdirコマンドでファイル名の一覧を取得する

システム要件を満たさないPCをWindows 11 2023 Update（23H2）にアップデートする方法

【Windows 10/11】えっ、UTF-8じゃなくてShift-JISで？ お手軽文字コード変換方法まとめ

PDFファイルにキーボードから直接文字入力する方法【本家Acrobat Reader編】

Excelの落とし穴「先頭のゼロ（0）」問題の対処法

【Windows 10/11】PCが数分で勝手にスリープするのを防ぐ

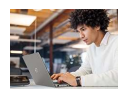
[ランキングをもっと見る](#)

あなたにおすすめの記事

- PR -



「ネットワークが分からない」状態からでも丸ごとサポート New!



社内ルールだけでは限界 有名無実化した「ローカル保存禁止」にどう対応？



オンプレのハードウェアも「サブスク」の時代へ コストや契約はどう変わる？

[@IT Specialへ](#)

ミドルの転職・AMBIの人気コンテンツ

- PR -



若手7割がスタートアップ転職に意欲 | AMBI（アンビ）



あなたの職務適性が15分でわかる | AMBI（アンビ）



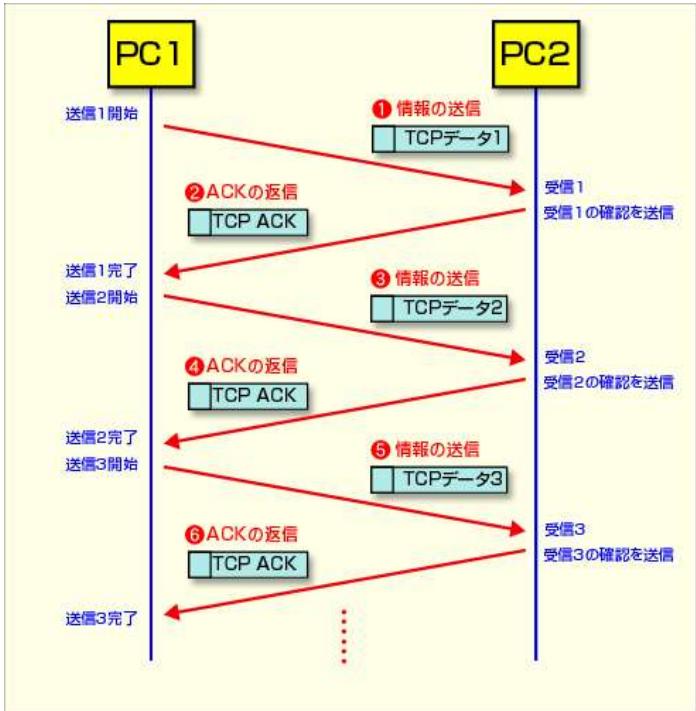
官公庁関連の厳選求人、多数掲載中！「ミドルの転職」

*1 常に固的なタイムアウト値を使用すると、さまざまな速度のネットワーク媒体に柔軟に対応するのが困難になる。例えば常にタイムアウトが1秒であると仮定すると、LANのような高速な媒体では再送処理が始まるまでの時間が長すぎ、パフォーマンスを最大限に生かすことができない。しかしWAN回線の場合は1秒では再送までの時間が短すぎ、再送パケットが無駄に数多く送信されることになるので、結果的にネットワークが混雑して、パフォーマンスも悪くなってしまう可能性がある。このような不具合を避けるため、動的に最適なタイムアウト値を決めるように実装するのが普通である。

なお、受信側が送信したACKパケットが、相手にまで正しく届いているかどうかを確認するためのパケットは送信されないが（つまり、ACKに対するACKを送信したりはしないということ）、この場合は、TCPパケットの送信側が再送処理を何度か行うので、受信側では、単にACKを再送信すればよい。このような仕組みのため、どこでパケットが消失しても正しくデータの送信が行えることになる。

ウィンドウ制御による効率のよいTCP送信

以上が基本的なTCPによる送信の原理であるが、これだけでは効率のよい通信を行うことはできない。例えば以下のように、TCPパケットを3つ送信するケースを考えてみよう。



効率の悪いACK応答
TCPパケットを1つ送信するごとに1つACKを受け取るとすると、遅延時間の大きなネットワークでは非常に効率が悪くなる。1パケットごとにネットワークの往復時間が必要になるからだ。

この例では、TCPパケットを1つ送信するごとに受信側から1つACKパケットを返信している。高速なLAN（イーサネット）回線なら特に問題ないだろうが、例えばパケットが相手に届くまで1秒もかかる、遠くの（そして細い）WAN回線だとしたらどうだろう。1つ目のTCPパケットが相手に届くまで1秒、そこからACKが戻ってくるまでに1秒（コンピュータ内部での送受信に要する時間は0秒と仮定する）、同様に、2つ目の送信とACKが完了するまでにまた2秒、そして3つ目の送受信が完了するまでにさらに2秒必要である。3つのTCPデータのパケットを送信するために計6秒必要となる。データ量が多い場合や、遅延時間がもっと大きい回線では、さらに時間が必要となる。

TCPでは、このような効率の悪い転送方法を改善するために、「ウィンドウ制御」という方式を採用している。詳細は後述するが、簡単に説明すると、これは複数のTCPパケットを連続して（ACK確認なしで）送信するという方式である。ウィンドウにはある決まったサイズがあり（プロトコルによっては、ウィンドウ・サイズをパケット数で数えたり、データのバイト数で数えたりする。TCPではバイト数単位となる）、このサイズの分だけ、（ACKを待たずに）データを連続して送信してもよい。具体的には次のように、送信したいデータを複数のTCPパケットに分け（1つのTCPパケットで送信できる

@IT eBook

解決！Python CSVファイル編

解決！Python CSVファイル編

「新えてやる！」の前に読むIT新説／徹底解説 vol.5

誰か、要件追加を止めてくれ！——「旭川医大の惨劇」徹底解説

Windowsファイル共有を“極める”ためのPowerShellコマンドレット基本集

目指せ、共有フォルダ管理の達人！ Windowsファイル共有を“極める”ためのPowerShellコマンドレット基本集

IT人材ゼロから始める中小企業のDXマニュアル

IT人材ゼロでDX!? お悩み中小企業のためのDX推進が分かる無料の電子書籍とは

一覧ページへ

注目のテーマ

AI for エンジニアリング

「サプライチェーン攻撃」対策

1P情シスのための脆弱性管理／対策の現実解

OSSのサプライチェーン管理、取るべきアクションとは

Microsoft & Windows 最前線 2023

システム開発ノウハウ【発注ナビ】 - PR -

Laravel

「Laravel」に強いシステム開発会社15社

React.js

コロナ禍の診察がスムーズに。基幹病院が開発した患者を守るシステム

React.js

「React.js」を使った開発で実績豊富な15社

IT

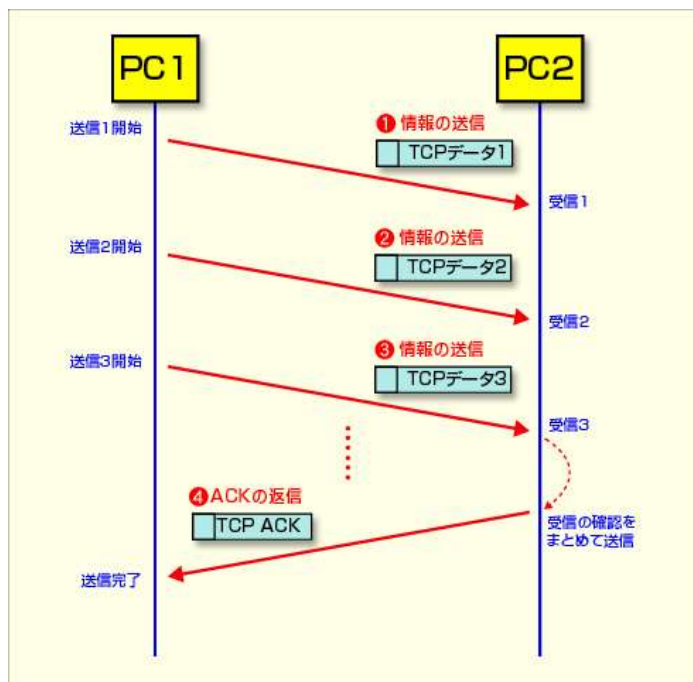
ページをフォロー 1.6万 フォロワー

@IT

9時間前

Microsoftは、起業家向けに生成AIを学べるトレーニングコンテンツをMicrosoft Learnで公開した。「アイデア発想」「プロトタイピングとMVP作成」「ビジネスモデル作成」の3つのフェーズで生成AIを活用する方法を学習できる。

データの最大サイズは、下位のIPプロトコルのパケットサイズによって制限される）、ACKを待たずにまとめて連続して送信する。そしてACKは、それらのデータに対して、まとめて1つだけ返送する。



TCPパケットの連続送信による送信効率の改善

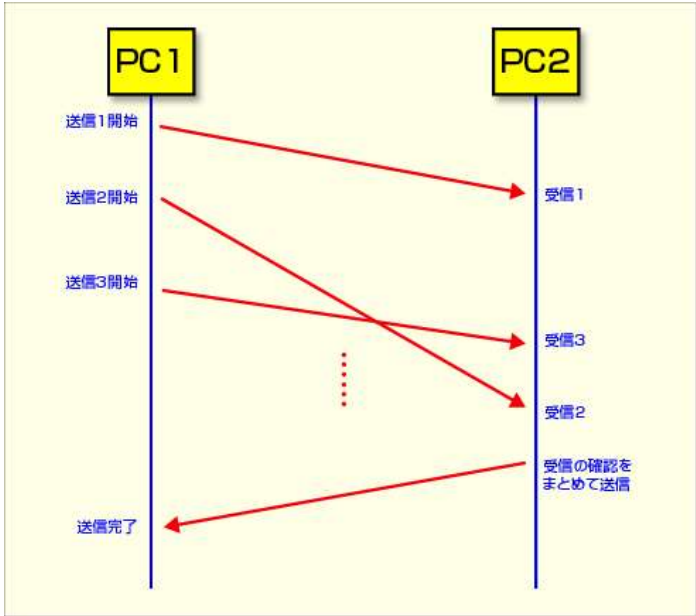
1パケットごとにACKを待たず、複数のパケットを受信してから1つだけACKを返すようにした例。送信側は、ACKを待つ必要がないので、連続してパケットを送信することができる。受信側も、1つだけACKを返せばよいので、手間も省けるし、ネットワークの使用帯域も最小限で済む。ACKパケットは非常に小さいので（TCPのヘッダ部分しかないパケットになる）、送信時間も最小限でよい。

この例では、3つのTCPデータ・パケットを送信し、それに対して1つだけACKを返している。ネットワークの遅延時間を1秒とすると、3つのパケットを連続して送信すれば、それらは1秒で受信側のマシンへ到着する（各パケットの送信間隔は、連続しているので0秒とする）。受信確認のACKが戻ってくるまでに1秒必要なので、結局、往復2秒で3つのパケットを送信することができる。先ほどの例では6秒かかっていたので、3倍高速である。パケットの数が多くなれば、その差はさらに広がる。すべてのパケットを送信し終わってから、ACKを1つだけ待てばよいのだから、ネットワークの「遅延時間」ではなく、ネットワークの「帯域幅」に大きく左右されることになる*2。

*2 ネットワークの特性を表す代表的な数値として、「遅延時間（latency）」と「帯域幅（bandwidth）」という2つのパラメータがある（これ以外にもさまざまなパラメータがある）。この2つは、似ているようだが、まったく異なるものである。一般に、高い（高い）帯域幅の回線（「速い回線」などと呼ばれることが多い）は遅延時間も短いことが多いが、常にそうであるとは限らない。例えば同じ100Mbpsのネットワークでも、ローカルの100BASE-TXのイーサネットならば1ミリ秒以内の遅延時間でパケットを相手に送信することができるが、100Mbpsの光ファイバで接続された海外のインターネット上のコンピュータならば数百ミリ秒は必要になる。通過するルータが多いときや、衛星回線などを経由する場合は、さらに遅延時間は長くなる。

ただしネットワークではパケットの往復が意味を持つことが多いので、性能評価には「RTT（Round Trip Time）」という、パケットの往復時間を使うことが多い。これを使うと例えば、理論的に実現可能なTCPの最大転送速度（スループット）は「ウィンドウ・サイズ÷RTT」と表せる。「ウィンドウ・サイズ」については後述する。狭帯域回線ではRTTの影響はあまり大きくないが、高速なブロードバンド回線ではRTTの影響は無視できなくなる。

このように、1パケットごとに受信確認を待たないようにすれば、回線帯域のほぼいっぱいまでデータを送信することができるので、ネットワークの帯域幅を効率よく使用することができる。ただしIPネットワークの特性上、送信したパケットが必ずしもその通りに到着するという保証はない。そのため、次のように順番が入れ替わって到着する可能性もある。



順番の入れ替わったパケット送信

IPプロトコルでは、送信したパケットの到着順序は保証されていない。ネットワークの環境によっては、後から送信したパケットの方が先に到着する可能性もある。このような場合でも、TCPでは問題なく処理できるように考慮されている。これにより無用な再送処理を防いでいる。

TCPでは、このような事態が発生しても問題がないように、ウィンドウ制御という方式を採用している。具体的には、送信するパケットには番号が付けられており、受信側では、パケットを番号順に並べ替えてから元のデータを復元し、それを上位のアプリケーションへ渡すようになっている。

なおIPパケットにも、フラグメント化されたパケットを順番通りに並べてから上位（TCPやUDP）へ渡す機能があるが、TCPのウィンドウ制御はそれとは独立した機能である。IPの場合は、フラグメント化したパケット（IPフラグメント）がすべて揃ってからIPパケットを再構成して上位へ渡していた（1つでも欠けていると再構成できない）。しかしTCPの場合は、分割されたパケットがすべて揃わなくても、ウィンドウの最初の方のパケットが到着さえすれば、その部分だけ先にACKを返してもよい（ACKを返すためには、ウィンドウの先頭から連続していなければならないため）。

シーケンス番号とウィンドウ制御

**C-Native**

クラウドシフトへの第一歩は、
「C-Native」から

伴走型支援

パッケージプラン

短期導入


C-Native Transformation Service



基礎から学ぶWindowsネットワーク 連載一覧

全 23 回

新しい連載記事が 7 件あります	
第16回	信頼性のある通信を実現するTCPプロトコル（3）
第15回	信頼性のある通信を実現するTCPプロトコル（2）
第14回	信頼性のある通信を実現するTCPプロトコル（その1）
第13回	データグラム通信を実現するUDPプロトコル

過去の連載記事が 11 件あります

Special

- PR -



データは「守りながら活用する時代」に



中堅中小企業の“ネットワーク課題”はこれで解決！ **New!**



オンプレのハードウェアも「サブスク」の時代へ コストや契約はどう変わる？



「守る」だけでは不十分 今どきのストレージには何が必要？



ローコードツールの現在地。AI、機械学習とのシナジーで新たな価値を生み出す **New!**



NTTデータと日本IBMがタッグ！ AIは仕事をどう変える？



「ほとんど誰も見ていない」社内ポータル、どう変えるべき？ **New!**



社内ルールだけでは限界 有名無実化した「ローカル保存禁止」にどう対応？

[@IT Special](#) へ

この記事に関連する製品／サービスを比較（キーマンズネット）

- まずネットワークの性質を十分に見極めよう！『ネットワーク管理』製品比較
- 構築したいネットワーク要件で大きく変わる『ルーター』の選び方
- 信頼性や可用性に対する取り組みは？『ネットワークスイッチ』製品比較
- 既存のネットワーク構成とマッチする？『WAN高速化』製品の選び方
- L4負荷分散とL7負荷分散どちらを重視？『ADC／ロードバランサ』製品一覧

印刷

通知

見る

Share

15

- @ITについて
- [お問い合わせ](#)
 - [広告について](#)
 - [採用広告について](#)
 - [利用規約](#)
 - [著作権・リンク・免責事項](#)
 - [サイトマップ](#)

- RSSについて
- [@ITのRSS一覧](#)

- アイティメディアIDについて
- [アイティメディアIDとは](#)

メールマガジン登録

@ITのメールマガジンは、もちろん、すべて無料です。ぜひメールマガジンをご購読ください。

申し込みページへ

ITmediaはアイティメディア株式会社の登録商標です。

[メディア一覧](#) | [公式SNS](#) | [広告案内](#) | [お問い合わせ](#) | [プライバシーポリシー](#) | [RSS](#) | [運営会社](#) | [採用情報](#) | [推奨環境](#)