Ciscoネットワーク演習１　　　 クラス　　　　番号　　　　氏名

3 - プロトコルとモデル

3.0 - 概要

3.0.1 - このモジュールを学ぶ理由

プロトコルは、標準組織によって作成された合意された規則のセットです。

モデルを使用すると、ネットワーク内のルールとその場所を視覚化することができます。

このモジュールでは、ネットワークプロトコルとモデルの概要を説明します。

3.0.2 - このモジュールで学ぶこと

**モジュールの目標**: ネットワークプロトコルにより、デバイスがローカルおよびリモートのネットワークリソースにアクセスする方法について説明

3.0.3 - 授業活動-通信システムの設計

省略します---何事にも通信(コミュニケーション)のためにルールが必要

3.1 - ルール

3.1.1 - ビデオ-バブル内のデバイス

ビデオ(約4分)を確認し、デバイス間の通信を可能にするプロトコルの組み合わせについて確認しましょう

3.1.2 - コミュニケーションの基礎

通信を行うには、デバイスが通信「手段」を知る必要があります。人々は多くの異なるコミュニケーション方法を使用してアイデアを交換します。

すべての通信方式には、共通する3つの要素が存在します。

* メッセージソース（**送信者**）  メッセージを送信する必要がある人、または電子デバイス。
* メッセージの宛先 (**受信者**)  宛先がメッセージを受信して解釈する。
* **チャンネル** 送信元から宛先に届くまでにメッセージが通過する[ ① **パス**]を提供。

3.1.3 - コミュニケーション プロトコル

通信は、フェイスツーフェイスで行う場合も、ネットワーク経由で行う場合もありますが、どちらの場合も[ ② **プロトコル**]と呼ばれるルールによって制御されます。手紙を送信するプロセスは、コンピュータネットワークで行われる通信に似ています。

3.1.4 - ルールの確立

プロトコルは、受信者が認識するメッセージを正常に配信するには、次の要件を考慮する必要があります。

* 識別された送信者と受信者
* 共通言語と文法
* 伝達の速さとタイミング
* 確認または確認応答の要件

3.1.5 - ネットワークプロトコルの要件

ネットワーク通信で使用されるプロトコルは、これらの基本的な特性の多くを共有します。

* メッセージの符号化
* メッセージのフォーマットとカプセル化
* メッセージ サイズ
* メッセージのタイミング
* メッセージの配信オプション

3.1.6 - メッセージの符号化

メッセージを送信するための最初のステップの 1 つは、[ ③ **符号化**]です。

符号化(エンコード)：情報を 1 つの形式から伝送で許容される別の形式に変換するプロセス

デコード：プロセスを逆にして情報を解釈します。

3.1.7 - メッセージのフォーマットとカプセル化

送信元から宛先にメッセージを送信するときは、決まった形式（構造）を使用する必要があります。メッセージの形式は、メッセージのタイプとメッセージの配信に使用されるチャネルによって異なります。

3.1.8 - メッセージ サイズ

メッセージのサイズにルールがあります。

3.1.9 - メッセージのタイミング

メッセージタイミングは、ネットワーク通信においても非常に重要です

[ ④ **フロー制御** ]：データ転送速度を管理するプロセスです。フロー制御は、送信できる情報の量と、それを配信できる速度を定義します。

**応答タイムアウト**：ネットワーク上のホストは、応答を待機する時間と、応答タイムアウトが発生した場合に実行するアクションを指定するネットワークプロトコルを使用します。

**アクセス方法**：デバイスが無線 LAN で送信する場合は、WLAN ネットワークインターフェイスカード(NIC)でワイヤレスメディアが使用可能かどうかを判断する必要があります。

3.1.10 - メッセージの配信オプション

ネットワーク通信には、コミュニケーションと同様の配信オプションがあります。

ユニキャスト ：情報が単一のエンドデバイスに送信されています。

マルチキャスト ：情報は 1 つ以上のエンドデバイスに送信されています。

[ ⑤ **ブロードキャスト**] ：情報がすべてのエンドデバイスに送信されます。

3.1.11 - ノードアイコンに関する注意事項

通常、ノードは円として表されます．

（Webテキストの図の場合、茶：送信者、緑：受信者、矢印：チャネル）

3.1.12 - 理解の確認 **Webサイトで「理解の確認」を実施してください．**

3.2 - プロトコル

3.2.1 - ネットワークプロトコルの概要

プロトコルは、ソフトウェア、ハードウェア、またはその両方でエンドデバイスと中間デバイスによって実装されます。

各ネットワークプロトコルには、通信のための独自の機能、形式、および規則があります。

[ ⑥ ]**：**複数のデバイスが 1 つ以上のネットワークを介して通信できるようになり ます。通信プロトコルの例：IP, TCP, HTTP

**セキュリティプロトコル：**データを保護し、認証、データ整合性、データ暗号化を提供します。セキュアプロトコルの例：SSH, SSL, TLS

**ルーティング プロトコル：**ルータはルート情報を交換し、パス情報を比較し、宛先への最適なパスを選択します。ルーティングプロトコルの例：OSPF,BGP

**サービス検出プロトコル：**デバイスまたはサービスの自動検出に使用されます。サービス検出プロトコルの例：DNS

3.2.2 - ネットワークプロトコル機能

ネットワーク通信プロトコルは、エンドデバイス間のネットワーク通信に必要なさまざまな機能を担います。

3.2.3 - プロトコル インタラクション

コンピュータネットワーク経由で送信されるメッセージは、それぞれ独自の機能と形式を持つ  
[ ⑦ 複数のプロトコル]を使用する必要があります。

例えばWeb通信の場合、HTTP、TCP、IP、Ethernetなど複数のプロトコルが使用されます。

3.2.4 - 理解したことを確認する **Webサイトで「理解したことを確認する」を実施してください．**

3.3 - プロトコル スイート

3.3.1 - ネットワーク プロトコル スイート

プロトコルスイートは、通信機能を実行するために必要な[ ⑧ 相互関連プロトコルのグループ]です。プロトコルスタックは、スイート内の個々のプロトコルがどのように実装されているかを示します。

3.3.2 - プロトコルスイートの進化

プロトコル スイートは、包括的なネットワーク通信サービスを提供することを目的として連携する一連のプロトコルです。

1970年代以降、いくつかの異なるプロトコルスイートがあった(現状はTCP/IP一択)

3.3.3 - TCP/IP プロトコルの例

TCP/IP プロトコルは、アプリケーション層、トランスポート層、およびインターネット層があります。ネットワークアクセス層には TCP/IP プロトコルはありません。

3.3.4 - TCP/IP プロトコル スイート

TCP/IPは、インターネットと今日のネットワークで使用されるプロコトルスイートです．TCP/IPには、ベンダーと製造業者にとって次の2つの重要な側面があります.

オープン標準プロトコルスイート：公衆が自由に利用可能であることを意味し、ハードウェアまたはソフトウェア上に任意のベンダーが使用することができます.

標準ベースのプロコトルスイート：ネットワーク業界によって承認され、標準化団体によって承認されたことを意味します．異なるメーカーの製品が正常に相互運用できるようになります.

3.3.5 - TCP/IP 通信プロセス

**Webサイトのアニメーションで、TCP/IP 通信プロセスについて確認しなさい**

3.3.6 - 理解の確認-プロトコルスイート **Webサイトで「理解の確認」を実施してください．**

3.4 - ネットワーク標準化団体

3.4.1 - オープンスタンダード

ネットワークコンポーネントにはさまざまな製造元があるため、すべて同じ規格を使用する必要があります。ネットワーキングでは、標準は[ ⑨ 国際標準化団体]によって開発されています。

オープンスタンダードは、相互運用性、競争、イノベーションを促進します。それらはまた、単一の会社の製品が市場を独占し、競争に対して不当な優位性を持つこと(ができないこと)を保証します。

標準組織は通常、オープンスタンダードの概念を開発し、推進するために設立されたベンダー中立の非営利組織です。

3.4.2 - インターネット標準

インターネットとTCP/IPプロトコルの標準を推進し作成するためのさまざまな組織があります

3.4.3 - 電子および通信規格

他の標準化団体は、有線または無線媒体を介して電子信号としてIPパケットを配信するために使用される電子および[ ⑩ 通信規格]を推進し、作成するための責任を持っています。

3.4.4 - ラボ-ネットワーキング標準の研究

省略します

3.4.5 - 理解の確認-標準化団体 **Webサイトで「理解の確認」を実施してください．**

3.5 - 参照モデル

3.5.1 - 階層型モデルを使用するメリット

実際のパケットが実際のネットワークを通過するのを見ることはできません。

このような状況では、モデルは便利です。

ネットワークの動作方法などの複雑な概念は、説明や理解が難しい場合があります。

このため階層型モデルは、ネットワークの操作を管理可能なレイヤーにモジュール化するために使用されます。

* 特定の層で動作するプロトコルには、その機能を定義する情報や上下の層とのインターフェイスが定められているため**[** ⑪**プロトコルの設計]**に役立ちます。
* 異なるベンダーの製品を組み合わせて使用できるので、競争が**[** ⑫**促進]**されます。
* あるレイヤーの技術や能力の変化が、上下の他のレイヤーに影響を与えません
* ネットワークの動作や機能を説明する**[** ⑬ **共通言語]**となります。

3.5.2 - OSI 参照モデル

OSI モデルは、各層で実行される機能やサービスの拡張リストを提供します。

TCP/IP プロトコルは、OSI モデルとTCP/IP モデルの両方で構成されています。

3.5.3 - TCP/IP プロトコル モデル

TCP/IP モデルは、TCP/IP スイート内のプロトコルの各層で発生する機能を記述するため、プロトコルモデルです。

3.5.4 - OSI と TCP/IP モデルの比較

TCP/IP プロトコルスイートを構成するプロトコルは、**[** ⑭ OSI 参照モデル **]**の観点から記述することもできます。

|  |  |  |  |
| --- | --- | --- | --- |
| レイヤ | OSI参照モデル | TCP/IPモデル | レイヤ |
| 7 | アプリケーション層 | アプリケーション層 | 4 |
| 6 | プレゼンテーション層 |
| 5 | セッション層 |
| 4 | トランスポート層 | トランスポート層 | 3 |
| 3 | ネットワーク層 | インターネット層 | 2 |
| 2 | データリンク層 | ネットワークアクセス層 | 1 |
| 1 | 物理層 |

3.5.5 - PacketTracer-TCP/IP および OSI モデルの実際の調査

省略します

3.6 - データのカプセル化

3.6.1 - メッセージのセグメント化

TCP/IP プロトコルスイートを使用して個々の IP パケットでデータをOSI 参照モデルと TCP/IP プロトコルモデルを知ることは、データがネットワーク上を移動するときにどのようにカプセル化されるかを知るときに便利です。

よりよい方法は，データはより小さく管理しやすいデータに**[** ⑮ **分割]**されて、ネットワークで送信することです．セグメンテーションは、ネットワークを介して送信するために、データのストリームをより小さな単位に分割するプロセスです。データネットワー送信するため、セグメンテーションが必要です。

メッセージのセグメント化には、主に 2 つの利点があります。

**高速化** -大きなデータストリームがパケットに分割されるため、通信リンクを接続することなく、大量のデータをネットワーク経由で送信できます。これにより、**[** ⑯ **多重化]**と呼ばれるネットワーク上でさまざまな会話がインターリーブされます。

**効率の向上** -ネットワークの障害やネットワークの輻輳により 1 つのセグメントが宛先に到達できなかった場合、データストリーム全体を再送信するのではなく、そのセグメントのみを  
**[** ⑰ **再送信]**する必要があります。

3.6.2 - シーケンシング

セグメンテーションと多重化を使用してネットワーク経由でメッセージを送信する上での課題は、プロセスに加わる複雑さのレベルです。

したがって、封筒内の情報には、受信者が適切な順序でページを再構成できるように、[ ⑱**シーケンス番号]**を含める必要があります。

3.6.3 - のプロトコル データ ユニット

特定の階層におけるデータの形態は、**[** ⑲**プロトコル データ ユニット]**（PDU）と呼ばれます。 プロセスの各階層の PDU には、それぞれの機能を表す異なる名称があります。

**データ**：アプリケーション層で使用されるPDUの総称

**セグメント(データグラム)**：トランスポート層のPDU

**パケット**：ネットワーク層のPDU

**フレーム**：データリンク層のPDU

**ビット**：メディアを介して物理的にデータを送信する時に使用される物理層のPDU

3.6.4 - カプセル化の例

ネットワークでメッセージを送信するときには、カプセル化プロセスが上から下に機能します。各層で、上位層の情報はカプセル化されたプロトコル内のデータと見なされます。たとえば、TCP セグメントは IP パケット内のデータと見なされます。

3.6.5 - カプセル化解除

このプロセスは、受信側のホストでは逆の順番で実行され、**[** ⑳**カプセル化解除]**と呼ばれます。

データはエンドユーザアプリケーションに向けてスタック内を移動するたびにカプセル化解除されます。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 送信データ | 対応する階層 | 受信データ |  |
| データ | アプリケーション～セッション層 | データ |
| セグメント | トランスポート層 | セグメント |
| パケット | ネットワーク層 | パケット |
| フレーム | データリンク層 | フレーム |
| ビット | 物理層 | ビット |

3.6.6 - 理解の確認-データのカプセル化　 **Webサイトで「理解の確認」を実施してください．**

3.7 - データ アクセス

3.7.1 - アドレス

ネットワーク内のメッセージを**[** ㉑ **セグメント化]**する必要があります。 しかし、それらのセグメント化されたメッセージは、適切に対処されていない場合、どこにも行きません。

**ネットワーク層の送信元アドレスと宛先アドレス**-元の送信元から最終宛先にIPパケットを配信します。最終宛先は、同じネットワーク上でもリモートネットワーク上でもかまいません。

**データリンク層の送信元アドレスと宛先アドレス**-データリンクフレームを 1 つのネットワークインターフェイスカード (NIC) から同じネットワーク上の別の NIC に配信します。

3.7.2 - レイヤ 3 論理アドレス

IP アドレスは、元の送信元から最終的な宛先に IP パケットを配信するために使用されるネットワーク層、つまりレイヤ 3 の論理アドレスです。

IP パケットには、次の 2 つの IP アドレスが含まれます。

**送信元IP アドレス**：パケットの元の送信元である送信側デバイスの IP アドレス。

**宛先IP アドレス**：パケットの最終宛先である受信デバイスの IP アドレス。

IP アドレスは 2 つの部分で構成されます。

**ネットワーク部（IPv4）またはプレフィックス（IPv6）**：IP アドレスがメンバーになっているネットワークを示すアドレスの左の部分。同じネットワーク上のすべてのデバイスは、アドレスの[ ㉒ **]**を持ちます。

**ホスト部（IPv4）またはインターフェイス ID（IPv6）**：ネットワーク上の特定のデバイスを識別するアドレスの残りの部分。この部分は、ネットワーク上のデバイスまたはインターフェイスごとに**[** ㉓**一意]**です。

3.7.3 - 同じネットワークのデバイス

送信元 IPv4 アドレスのネットワーク部分と宛先 IPv4 アドレスのネットワーク部分が同じであるため、送信元と宛先は**同じネットワーク**上にあることに注意

3.7.4 - データリンク層アドレスの役割：同じ IP ネットワーク

IP パケットの送信側と受信側が同じネットワーク上にある場合、データリンクフレームは受信デバイスに直接送信されます。

MAC アドレスは、イーサネット NIC に物理的に埋め込まれています。

3.7.5 - リモートネットワーク上のデバイス

デバイスがリモートネットワーク上のデバイスと通信しているときに、ネットワーク層アドレスとデータリンク層アドレスの役割は？

3.7.6 - ネットワーク層アドレスの役割

送信元 IPv4 アドレスと宛先 IPv4 アドレスのネットワーク部分が異なるネットワーク上にあることに注目

3.7.7 - データリンク層アドレスの役割:異なる IP ネットワーク

IP パケットの送信側と受信側が異なるネットワーク上にある場合、送信側のネットワーク内でホストが直接到達できないため、イーサネットデータリンクフレームを宛先ホストに直接送信することはできません。イーサネットフレームは、ルータまたは**[** ㉔ **]**と呼ばれる別のデバイスに送信する必要があります。

3.7.8 - データリンクアドレス

データリンクレイヤ 2 物理アドレスの役割は異なります。データリンクアドレスの目的は、データリンクフレームを 1 つのネットワークインターフェイスから同じネットワーク上の別のネットワークインターフェイスに配信することです。

IPパケットがホストからルーター、ルーターからルーター、そして最後にルーターからホストに移動するとき、途中の各ポイントでIPパケットが新しいデータリンクフレームにカプセル化される。

3.7.9 - ラボ-Wiresharkをインストールする

省略　実習時にインストールします

3.7.10 - ラボ-Wiresharkを使用したネットワークトラフィックの表示

実習時に実施

3.7.11 - 理解したことを確認する　 **Webサイトで「理解の確認」を実施してください．**

3.8 - モジュール演習とクイズ

3.8.1 - このモジュールで学んだこと

3.8.2 - モジュールクイズ-プロトコルとモデル

**Webサイトで「モジュールクイズ」を実施してください．**