**Camada de Aplicação**

* **Aplicação, Apresentação e Sessão**

*Aplicação:*

A camada de aplicação é a que está mais associada ao usuário final nos modelos OSI e TCP/IP. É a camada de interface entre os aplicativos utilizados para se comunicar e a rede na qual as mensagens são transmitidas. Os protocolos desta trocam dados entre programas em execução nos hosts de origem e de destino. Os protocolos de camada de aplicação são muitos e novos sempre são desenvolvidos. Exemplos, HTTP (Protocolo de Transferência de Hipertexto), FTP (Protocolo de Transferência de Arquivo), TFTP (Protocolo de Transferência de Arquivo Trivial), IMAP (Protocolo de Acesso a Mensagem da Internet) e DNS (Sistema de Nomes de Domínio).

*Apresentação e Sessão:*

A camada de apresentação tem três funções principais:

* Formatar ou apresentar os dados no dispositivo de origem em um formato compatível para recebimento no dispositivo de destino.
* Comprimir dados na transmissão e descompactar no dispositivo de destino (recepção).
* Criptografar os dados para transmissão e descriptografar na recepção.

A camada de apresentação prepara os dados para a camada de aplicação e define padrões para formatos de arquivo. Já a camada de sessão cria e mantém comunicação entre os aplicativos de origem e de destino. A camada de sessão trabalha na troca de informações para iniciar a comunicação, mantê-la ativa e reiniciar sessões interrompidas ou ociosas por um tempo considerado longo.

* **P2P (Point-to-Point)**

 No modelo cliente/servidor o dispositivo que solicita as informações é denominado cliente e o dispositivo que responde à solicitação é denominado servidor. Os processos de cliente e servidor são considerados na camada de aplicação. O cliente inicia ao solicitar dados do servidor que responde ao enviar um ou mais fluxos de dados ao cliente. Os protocolos da camada de aplicação descrevem o formato das solicitações e respostas entre clientes e servidores. Executa a transferência real de dados e além disso, também, pode necessitar da autenticação do usuário e a identificação de um arquivo de dados a ser transferido. Um exemplo de modelo de rede cliente/servidor é o serviço de e-mail de um ISP para enviar, receber e armazenar e-mail. Em uma rede ponto-a-ponto (P2P) os dados são acessados ​​a partir de um dispositivo sem o uso de um servidor dedicado. O P2P engloba duas partes: redes P2P e aplicativos P2P. Em uma rede P2P, dois (ou mais) computadores são conectados via rede e compartilham recursos (como impressoras e arquivos) sem ter um servidor dedicado. Cada dispositivo final conectado (denominado peer, par) pode funcionar como servidor e cliente. Um computador pode assumir a função de servidor para uma transação e, ao mesmo tempo, ser cliente para outra. Em uma troca ponto-a-ponto, os dois dispositivos são considerados iguais no processo de comunicação.

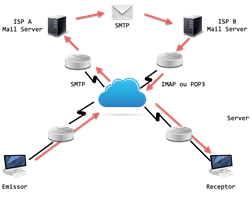
*Aplicações:*

Uma aplicação P2P possibilita que um dispositivo aja tanto como cliente quanto como servidor na mesma comunicação. Assim, nesse modelo todo cliente é um servidor e todo servidor é um cliente. Os aplicativos P2P exigem que cada dispositivo final provejam uma interface de usuário e execute um serviço em segundo plano. Alguns aplicativos P2P utilizam um sistema híbrido em que o compartilhamento de recursos é descentralizado, mas os índices que apontam para as localizações dos recursos são armazenados em um diretório centralizado. Em tal sistema cada par acessa um servidor de indexação para obter a localização de um recurso armazenado em outro par.

* **Protocolos Web**

*E-mail:*

A Figura a seguir mostra os protocolos de e-mail.



*E-mail – SMTP (Simple Mail Transfer Protocol):*

A Figura a seguir ilustra o protocolo SMTP (porta 25).



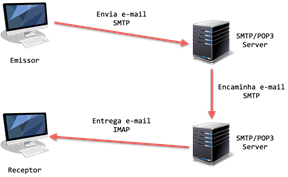
*E-mail - POP3 (Post Office Protocol 3):*

A Figura a seguir mostra o protocolo POP-3.



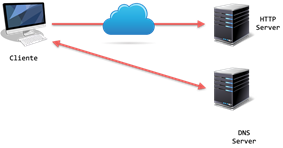
*E-mail – IMAP (Internet Message Access Protocol):*

Diferente do POP3, o usuário ao se conectar a um servidor compatível com IMAP, as cópias das mensagens são carregadas no aplicativo cliente. As mensagens originais são mantidas no servidor até serem excluídas (de forma manual). Os usuários visualizam cópias das mensagens no software cliente de e-mail. A Figura a seguir ilustra o protocolo IMAP.



*HTTP (Hypertext Transfer Protocol):*

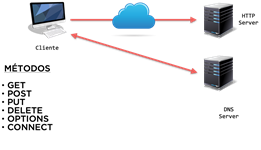
A Figura a seguir mostra o servidor HTTP.



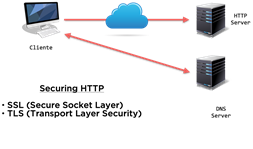
A Figura a seguir ilustra o formato da URL (*Uniform Resource Locator*).



O protocolo HTTP é de solicitação/resposta. Um navegador da web ao enviar uma solicitação a um servidor da web, o HTTP especifica os tipos de mensagem usados para essa comunicação. A Figura a seguir mostra os tipos de mensagem comuns.



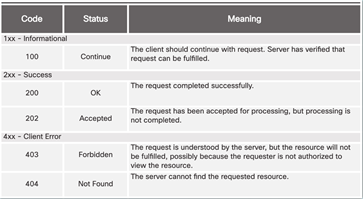
​Para navegar na Internet de forma segura é utilizado o protocolo HTTP Secure (HTTPS). O HTTPS utiliza autenticação e criptografia para proteger os dados ao trafegarem entre o cliente e o servidor. O HTTPS utiliza o mesmo processo de resposta do servidor de solicitação do cliente que o HTTP, mas o fluxo de dados é criptografado com Secure Socket Layer (SSL) antes de trafegar pela rede (STALLINGS, 2015). A Figura a seguir ilustra a comunicação HTTPS.​



A Figura a seguir mostra os códigos de status do HTTP.



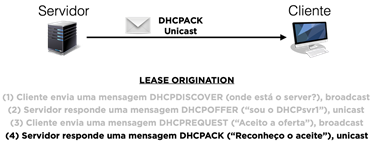
A Tabela a seguir mostra os detalhes dos códigos de status do HTTP.



* **Serviços**

*DHCP (Dynamic Host Configuration Protocol):*

A Figura a seguir ilustra as mensagens de descoberta do servidor DHCP.

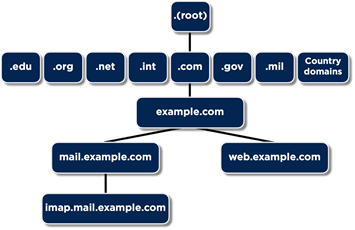


A Figura a seguir mostra mensagens do servidor DHCP para renovar a solicitação do cliente.

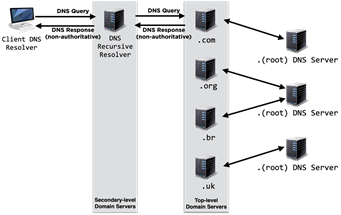


DNS *(Domain Name System):*

A Figura a seguir ilustra hierarquia de servidores DNS.



A Figura a seguir mostra mensagens de resolução de nomes com servidores DNS.

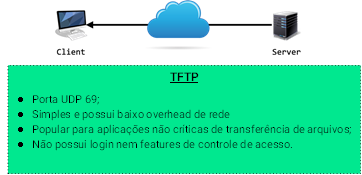


*FTP e TFTP:*

A Figura a seguir ilustra as mensagens entre cliente e servidor FTP (File Transfer Protocol).



A Figura a seguir mostra uma transferência mais simples (trivial) com TFTP (Trivial File Transfer Protocol).



A Figura a seguir ilustra as mensagens SMB.

