UniFil
EDUCAÇÃO A DISTÂNCIA

Unidade 4

A camada de Aplicação e Conceitos de Segurança da Informação

Abertura

Apresentação

Esta unidade apresenta a camada de aplicação, a camada que temos mais interações no nosso dia a dia, e muitas vezes nem nos damos conta. Muitos serviços são essenciais e "invisíveis" para que outros serviços possam ser executados. Se você acessa a Internet, já buscou alguma coisa na web, ou precisou enviar um e-mail, ou teve que fazer uma transferência de arquivos, são exemplos de aplicações que rodam na camada de aplicação.

Ainda nesta unidade, finalizamos vendo alguns conceitos de Segurança da Informação, e destaco a importância deles, ainda mais agora, numa era em que estamos digitalizando muitos serviços e processos e fazendo uso intenso da computação nas nuvens.

Objetivos

- Conhecer os serviços da camada de aplicação;
- Conhecer os conceitos essenciais de Segurança da Informação;
- Compreender a divisão entre camadas e as responsabilidades de cada camada.

Conteúdo programático

Aula 01 – A Camada de Aplicação

Aula 02 – Conceitos básicos de Segurança da Informação

A Camada de Aplicação

Nesta aula, vamos discutir os aspectos da camada de aplicação, pois é nesta camada que temos o serviço que um usuário terá acesso e disponibilidade para interagir. Este usuário pode ser uma pessoa ou outra máquina, mas é aqui que temos a interface na sua mais alta camada. A camada de aplicação é responsável por fornecer serviços.

Como discutido anteriormente, geralmente, no padrão TCP/IP encontramos predominantemente o paradigma cliente/servidor, isto é, o cliente faz uma requisição ao servidor, onde geralmente está o serviço que queremos ter acesso. Entretanto, vimos também que os endereços são na sua maioria escritos de forma decimal com pontos no IPv4 e hexadecimal com separação em usando dois pontos (:). Seria praticamente impossível para nós decorarmos esses valores toda vez que quiséssemos acessar uma página web, por exemplo. Eis aí, portanto, nossa primeira aplicação a ser estudada: ***Domain Name System - DNS***.

O DNS é um serviço que dá suporte aos outros serviços, pois como vimos, a ideia dele é traduzir endereços IP em nomes entendíveis por nós. Por exemplo, quando acessamos uma página web, nós escrevemos: "www.google.com", mas sabemos que para enviar os dados na rede, precisamos dos endereços IP, e como descobrir o endereço IP do "google.com"? Executei um simples comando de ping, como pode ser visto abaixo.

```
➜ ~ ping www.google.com
PING www.google.com (142.251.128.100): 56 data bytes
64 bytes from 142.251.128.100: icmp_seq=0 ttl=119 time=12.440 ms
64 bytes from 142.251.128.100: icmp_seq=1 ttl=119 time=15.544 ms
64 bytes from 142.251.128.100: icmp_seq=2 ttl=119 time=12.684 ms
64 bytes from 142.251.128.100: icmp_seq=3 ttl=119 time=12.817 ms
64 bytes from 142.251.128.100: icmp_seq=4 ttl=119 time=12.711 ms
```

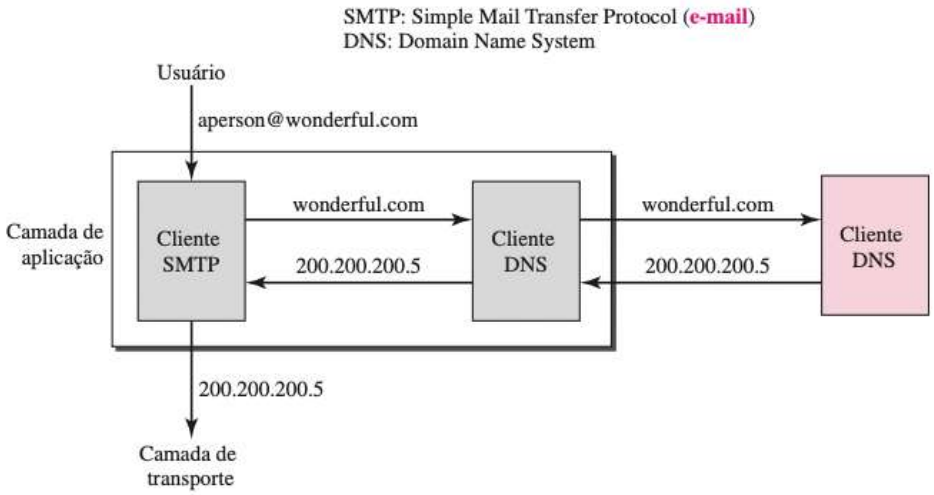
O programa ping envia mensagens de solicitação de ECO no protocolo ICMP. Geralmente esse comando é usado para testar o nome do computador e o endereço IP. Como o endereço "www.google.com" consegue ser encontrado, ele me retorna a mensagem com o IP da máquina que está hospedando o serviço, no caso: 142.251.128.100. Esse nome é o que chamamos de **domínio** e ele fica atrelado a um endereço IP.

Indicação de Vídeo

Neste vídeo curto criado pelo ICANN, eles explicam o que é o nome de domínio. What Is a Domain Name?

E o DNS é um serviço essencial para muitos outros serviços que executam na Internet, como o próprio e-mail, que veremos mais adiante. O protocolo de e-mail (SMTP - Simple Mail Transfer Protocol) se utiliza do DNS para resolver para onde o e-mail deve ser encaminhado. A ideia do DNS está ilustrada na Figura 1.

Figura 1: Exemplo de uso do serviço DNS pelo SMTP

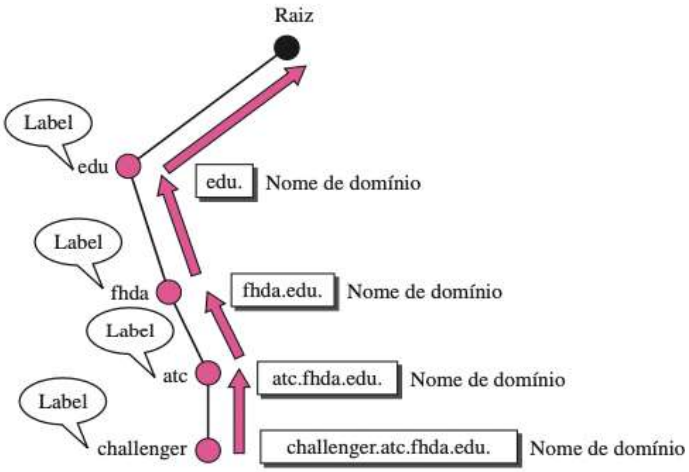


Fonte: FOROUZAN, 2009.

No início, quando a rede tinha poucos segmentos, um arquivo contendo o nome e o endereço resolvia o problema. Mas com o aumento contínuo da rede, e as alterações constantes de nome e endereço, essa abordagem se tornou inadequada. Hoje, o DNS funciona de maneira hierárquica, dividindo as informações em partes menores e armazenando cada uma dessas partes em um computador diferente. Assim, o host que precisar de mapeamento poderá entrar em contato com o computador mais próximo que possui as informações necessárias.

Os nomes de domínio funcionam de maneira hierárquica, segue um modelo de árvore, mas sempre visualizada de maneira invertida, com a raiz na parte superior e só pode ter no máximo 128 níveis. Com esse modelo hierárquico, podemos definir e dar sentido para as partes, assim, o nome de domínio poderia ter na sua primeira parte o país, na segunda à natureza da organização, na terceira o nome da organização, e assim, irmos adicionando o que chamamos de **subdomínios**. Cada parte, podemos chamar de **label**.

Figura 2: Labels e nomes de domínio.

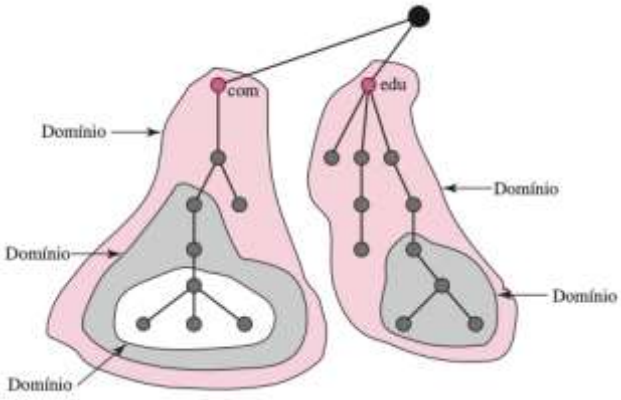


Fonte: FOROUZAN, 2009.

Se você já estiver familiarizado com o conceito de árvores em estrutura de dados, aqui é um exemplo da sua aplicação também. Assim como todo nó é a raíz da sua subárvore, o mesmo se aplica aqui nos domínios. Um domínio é uma subárvore do espaço de nomes de domínios. O nome do domínio é o do nó na parte superior da subárvore. E um domínio pode ser dividido em outros domínios (ou subdomínios, como são algumas vezes chamados), a Figura 3 ilustra essa ideia.

Os domínios são organizados pela ICANN e outras organizações regionais. No Brasil temos o Registro.br como responsável pelo registro no domínio **.br**. Novos domínios são adicionados ao DNS por meio de uma entidade registradora, uma entidade comercial homologada pela ICANN. Esse serviço tem uma taxa que depende de entidade para entidade, e assim como no IP, esse nome de domínio solicitado deve ser exclusivo.

Figura 3: Domínios e subdomínios



Fonte: FOROUZAN, 2009.

Indicação de Leitura

Para maiores detalhes, acessem a RFC 1034. RFC 1034 - DOMAIN NAMES - CONCEPTS AND FACILITIES

Videoaula 1

Utilize o QR Code para assistir!

Agora, assista ao vídeo em que o professor aborda todo esse conteúdo introdutório e explica com mais detalhes o funcionamento do DNS.

Uma aplicação que se tornou popular e bem comum hoje é o **webmail**. Praticamente não vivemos sem e-mail, apesar de não usá-lo tanto, todos temos contas em algum serviço de webmail. Me lembro ainda, no início do serviço de correio eletrônico da Google quando era necessário ser indicado para ter uma conta no domínio gmail. Hoje, basta acessar o site e criar uma conta no gmail.

O webmail é uma forma simples de visualizar o email através da web pelo navegador do usuário, mas por trás do que acontece, ainda é em grande parte como acontecia no início dos correios eletrônicos. É necessário a utilização de alguns protocolos e componentes. Os protocolos são: **SMTP (*Simple Mail Transfer Protocol*)**, **POP3 (*Post Office Protocol v3*)** e **IMAP4 (*Internet Mail Access Protocol v4*)**. Os componentes são: **UA (*User Agent*), MTA (*Message Transfer Agent*)** e **MAA (*Message Access Agent*)**.

Indicação de Leitura

Para maiores detalhes, acessem as RFCs dos protocolos.

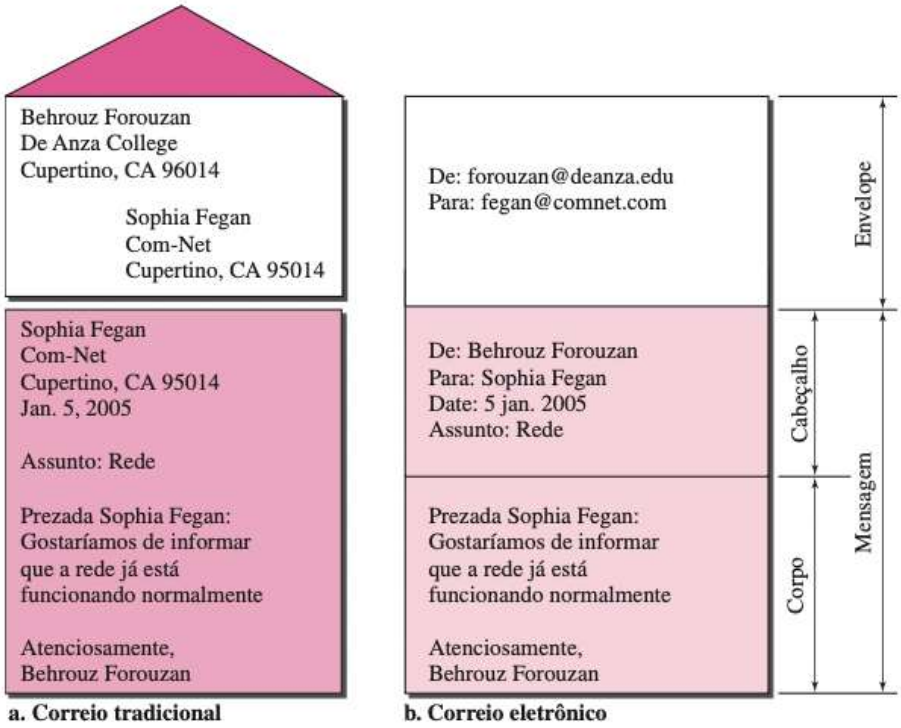
RFC 821 - SIMPLE MAIL TRANSFER PROTOCOL

RFC 1081 - Post Office Protocol - Version 3

RFC 3051 - INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4

O UA é o componente que fornece os serviços relacionados a mensagem como criar, ler, responder, encaminhar e receber. Geralmente é um programa de computador, que pode ou não possuir uma interface gráfica. Ela possui uma caixa de entrada (*inbox*) e uma caixa de saída (*outbox*). O formato de um e-mail é muito parecido com o formato de uma carta tradicional, vide Figura 4.

Figura 4: Formato de um e-mail



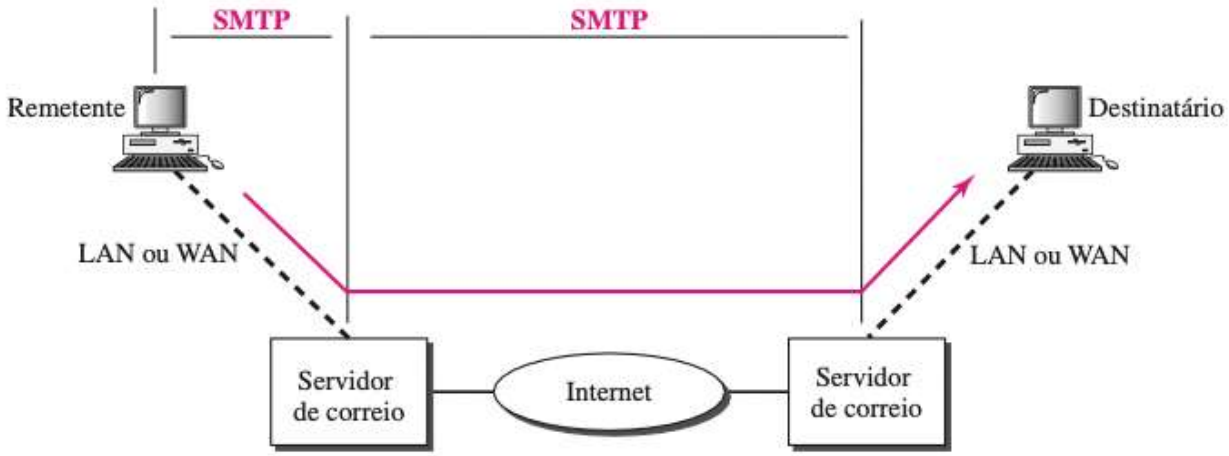
Fonte: FOROUZAN, 2009.

Assim como nas cartas tradicionais, no correio eletrônico precisamos de um endereço que seja único e exclusivo, senão não é possível encontrar o remetente. No e-mail o endereço consiste em duas partes: uma parte local e um nome de domínio, separados pelo símbolo @, por exemplo, "mario@email.com". Não é possível existir dois endereços com o mesmo formato.

No início, o correio eletrônico era para troca de mensagens textuais e não muito compridas. Com o passar do tempo, o conteúdo da mensagem trocada foi alterando, e hoje, sabemos que é muito comum enviar arquivos dentro de uma mensagem. Isto é possível, pois o **MIME (*Multipurpose Internet Mail Extensions*)** foi adicionado como protocolo complementar dando suporte para dados não ASCII.

A transferência das mensagens é realizada pelo segundo componente, o **MTA (Message Transfer Agents)**. Seguimos o paradigma cliente/servidor, sendo necessário dois atores para executar, e o protocolo conversado entre eles é o SMTP. A transferência ocorre como visto na Figura 5.

Figura 5: Funcionamento do SMTP



Fonte: FOROUZAN, 2009.

Como demonstrado na Figura 6, o SMTP é aplicado na comunicação entre o cliente e o servidor, e num momento posterior entre o servidor e outro servidor de correio. A comunicação entre um cliente MTA e um servidor MTA é usando comandos e respostas do SMTP. A ideia do fluxo da comunicação pode ser vista abaixo (FOROUZAN, 2009).

========== Connection Establishment ==========

220 mta13.adelphia.net SMTP server ready Fri, 6 Aug 2004 . . .

HELO mail.adelphia.net

250 mta13.adelphia.net

========== Mail Transfer ========== MAIL FROM: forouzanb@adelphia.net

250 Sender <forouzanb@adelphia.net> Ok

RCPT TO: forouzanb@adelphia.net

250 Receiver <forouzanb@adelphia.net> Ok

DATA

354 Ok Send data ending with <CRLF>.<CRLF>

From: Forouzan TO: Forouzan

This is a test mensagem

to show SMTP in action.

•

========== Connection Termination ==========

250 Message received: adelphia.net@mail.adelphia.net

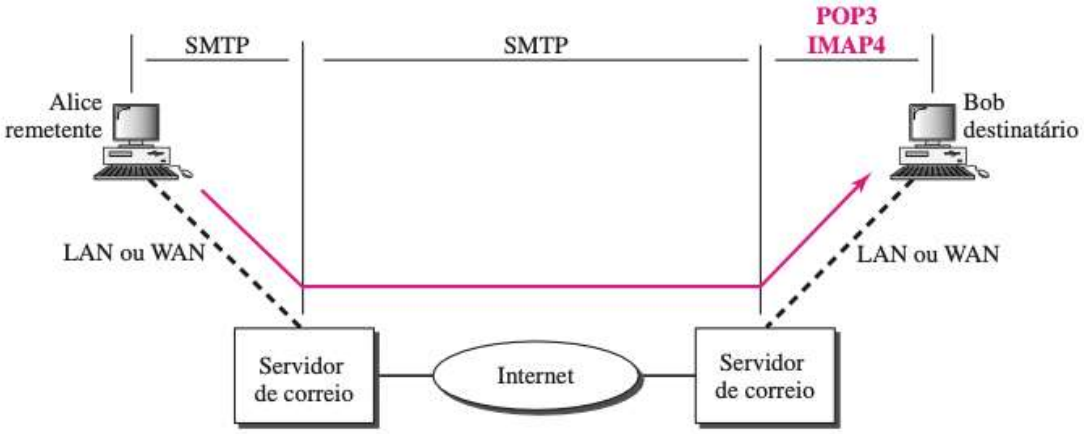
QUIT

221 mta13.adelphia.net SMTP server closing connection

Connection closed by foreign host.

O terceiro componente é o **MAA (*Message Access Agent*)**, pois o SMTP faz a ação de *push*, ou seja, ele apenas envia a mensagem. É necessário outro protocolo que faça o sentido contrário, que "puxa" as mensagens do servidor para o cliente. Existem dois protocolos, POP3 e IMAP4.

Figura 6: POP3 e IMAP4 na comunicação para puxar as mensagens



Fonte: FOROUZAN, 2009.

Como o POP3 surgiu antes, é um protocolo simples e limitado. Ele abre uma conexão entre o cliente e o servidor, porta 110 usando TCP. Após a autenticação do usuário, é possível listar as mensagens e baixá-las. O protocolo ainda possui dois modos: ***keep*** e ***delete***. No modo *delete*, assim que a mensagem é baixada no computador cliente, ela é removida do servidor. O modo *keep, por* sua vez, não apaga a mensagem, mantendo uma cópia no servidor e outra no cliente.

O IMAP4 é um avanço natural do protocolo, pois adiciona funcionalidades que não eram possíveis no POP3 como: organização em pastas, verificar o cabeçalho antes de baixar a mensagem, baixar parcialmente um email, procurar por um conteúdo sem baixar as mensagens, entre outros recursos. Muitas vezes, temos coexistência entre eles em diversos servidores e os programas de email aceitam e trabalham com os dois protocolos.

Outra aplicação muito utilizada é a transferência de arquivos, e o TCP/IP utiliza o **FTP (File Transfer Protocol)** para copiar arquivos de um host para outro, na porta 20 e 21. O FTP utiliza duas portas, uma para controle de conexão (21) e outra para a conexão e troca de dados efetivamente (20). A conexão de controle permanece estabelecida durante toda a interatividade de uma sessão FTP. Uma conexão de dados será estabelecida e posteriormente encerrada para cada arquivo transferido.

O FTP pode transferir arquivos de texto no formato ASCII, sendo que cada caractere é codificado usando 7 bits ASCII, e cada ponta da comunicação converte os caracteres ASCII em

caracteres de representação próprios. Outro formato possível de enviar via FTP são arquivos binários, que são transmitidos através de fluxos contínuos de bits (*streams*), não sendo necessário a conversão pelos participantes da troca de informação. Além dos tipos de arquivos, o FTP possui três modos de transmissão: fluxo contínuo, de blocos e comprimido. O fluxo contínuo é o modo padrão, onde o FTP passa para o TCP os dados na forma de um fluxo contínuo de bytes, passando a responsabilidade para o TCP de dividir os dados em segmentos de tamanho apropriado. No modo de blocos, os dados são entregues do FTP para o TCP em blocos. Nesse caso, cada bloco é precedido por um cabeçalho de três bytes. Por fim, o modo comprimido é utilizado quando o arquivo é muito grande e os dados são comprimidos.

Indicação de Leitura

Para maiores detalhes, acessem a RFC 765.

RFC 765 - FILE TRANSFER PROTOCOL

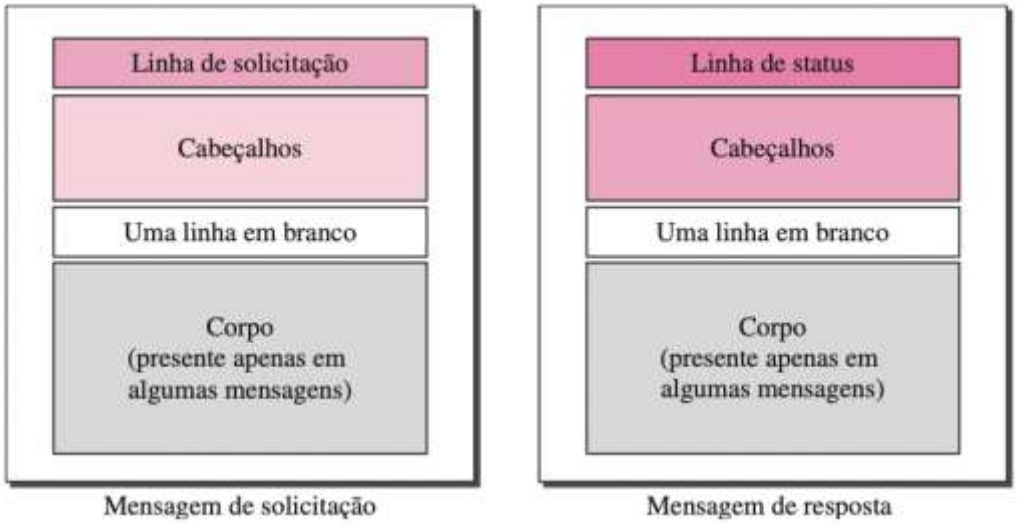
Videoaula 2

Utilize o QR Code para assistir!

Agora, assista ao vídeo em que o professor aborda as aplicações de correio eletrônico e transferência de arquivos.

O protocolo mais largamente conhecido e utilizado hoje em dia é o **HTTP (Hypertext Transfer Protocol),** que é utilizado para acessar dados na Web. Também trabalha num paradigma cliente/servidor, geralmente o cliente é um navegador, e a porta mais conhecida é a 80. Ele trabalha com solicitação e resposta, ou seja, o cliente envia uma requisição e recebe uma resposta do servidor. A Figura 7 mostra as mensagens de solicitação e resposta.

Figura 7: Mensagens de solicitação e resposta do HTTP

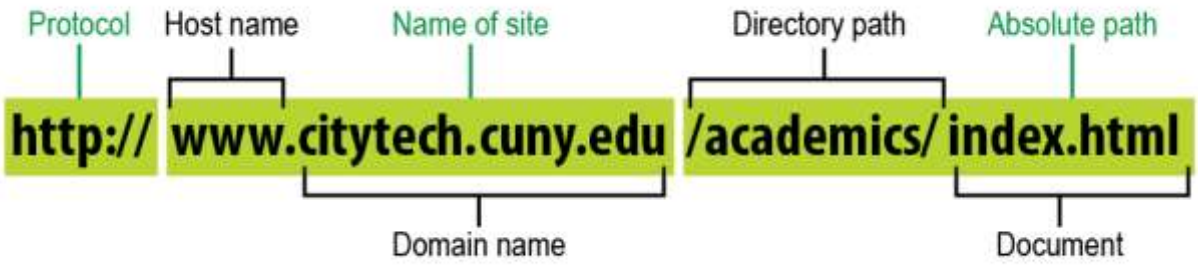


Fonte: FOROUZAN, 2009.

Na **Linha de solicitação**, é formada pelo tipo de solicitação, que chamamos de métodos HTTP, podendo ser: **GET, HEAD, POST, PUT, TRACE, CONNECT** e **OPTION**. Cada um dos métodos tem uma função específica, por exemplo, o GET é para solicitar um recurso, o HEAD solicita informações sobre o recurso, POST envia informações para o servidor, OPTION solicita detalhamento sobre opções disponíveis no servidor relacionados ao recurso. A versão é outro campo utilizado, no caso HTTP 1.1 é a mais atual, e existem propostas para o HTTP 2.0.

Além do tipo de solicitação, é necessário localizar o recurso que se quer acessar, e isto é feito através da **URL (*Uniform Resource Locator*),** um padrão para a especificação de qualquer tipo de informação na Internet. Uma URL é formada pelo protocolo, host, nome de domínio e diretório. Ainda é possível ter a porta associada ao recurso, neste caso, haveria uma separação com dois pontos (:) logo após o domínio. A Figura 8 mostra as partes que formam uma URL.

Figura 8: URL



Fonte: https://stuyhsdesign.files.wordpress.com/2015/09/urlparts.gif. Acesso em: 6 jun. 2022.

O Cabeçalho permite fornecer informações gerais sobre a mensagem e pode ser utilizado tanto em uma mensagem de solicitação quanto de resposta. Por exemplo, o cliente pode

solicitar que um documento seja enviado em um formato especial e isto pode ser especificado no cabeçalho através do campo Accept. Além disso, muitas outras informações são passadas, como o tamanho do documento, o tipo do documento, quando foi modificado pela última vez, localização e até permissões.

Indicação de Vídeo

Neste vídeo, vocês aprenderão um pouco mais sobre o HTTP com o casal do Código Fonte TV.
HTTP // Dicionário do Programador

A Figura 10, retirada de (FOROUZAN, 2009) mostra o acesso a um documento. Usamos o comando GET para obter uma imagem no caminho /usr/bin/image1. A linha de solicitação exibe o comando (GET), a URL e a versão do HTTP (1.1). O cabeçalho apresenta duas linhas informando que o cliente pode aceitar imagens nos formatos GIF ou JPEG. A mensagem de solicitação não tem um corpo. A mensagem de resposta contém a linha de status e quatro linhas de cabeçalho. As linhas de cabeçalho definem a data, o servidor, a versão MIME e o tamanho do documento. O corpo do documento segue logo após o cabeçalho.

Figura 10: Exemplo de solicitação HTTP

Fonte: FOROUZAN, 2009.

E assim finalizamos mais uma aula, apresentando algumas das aplicações que mais se destacam na camada de aplicação. Vale mencionar que existem outros protocolos, alguns abertos com maior interoperabilidade e outros protocolos fechados, para domínios específicos. A camada de aplicação é aquela que temos maior contato no nosso dia a dia,

apesar de não lidarmos diretamente com ela, o nosso navegador ou qualquer outro programa que usamos e esteja conectado na rede, utiliza algum tipo de protocolo para prover o serviço que estamos usando.

Videoaula 3

Utilize o QR Code para assistir!

Agora, assista ao vídeo em que o professor aborda o HTTP e encerra mais uma aula do nosso curso.

Segurança da Informação

Discutiremos aqui, alguns conceitos de Segurança da Informação e como é possível levar bits de um lado para o outro com mais segurança. Vamos explorar um o conceito de criptografia, uma ciência baseada em álgebra que tem como objetivo transformar as mensagens e torná-las seguras. Quando estamos lidando com comunicação em redes de computadores, formada por diversos tipos de componentes, sempre existe um lado mais sensível aos ataques. A criptografia pode fornecer diversos aspectos de segurança relacionados à troca de mensagens através de redes. Confidencialidade, integridade, autenticação e o não repúdio são os aspectos que conseguimos cobrir com a criptografia, dependendo da forma que a utilizamos.

Criptografia significa "escrita secreta", de origem grega. Nós usamos o termo para definir a ciência que estuda a transformação das mensagens de modo seguro e imune a ataques. Chamamos de **texto claro** o que desejamos enviar, e **texto cifrado** o resultado da criptografia e usualmente o que enviamos. O emissor utiliza uma **chave** em uma **cifra,** este é o ato de criptografar, e então envia o texto cifrado. O receptor recebe o texto cifrado e precisa realizar a decriptografia do texto cifrado, utilizando uma chave também, para então ter o **texto claro** novamente. Este processo pode ser visto na Figura 10.

Figura 10. Componentes da criptografia

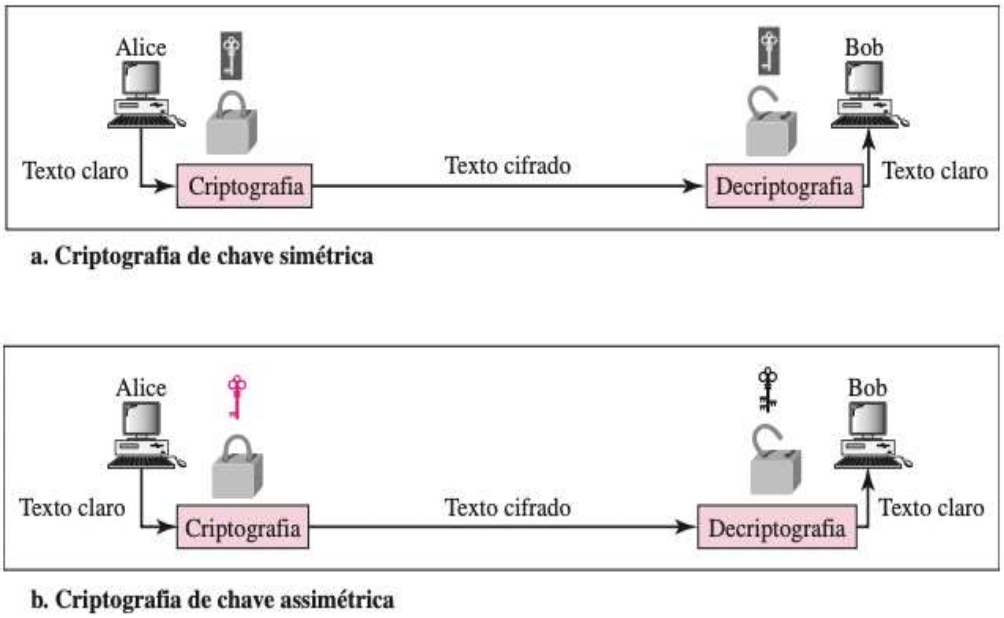
Fonte: FOROUZAN, 2009.

Podemos dividir todos os algoritmos de criptografia em dois grupos: algoritmos de criptografia de chave simétrica (também chamadoschave secreta) e algoritmos de criptografia assimétrica (também denominados chave-pública). Na criptografia de chave simétrica utilizamos uma chave que é compartilhada entre ambos, emissor e receptor. Ou seja, a mesma chave que criptografa é a mesma que descriptografa também.

Já na criptografia assimétrica, temos duas chaves: uma privada e outra pública. A chave privada, como o nome sugere, é individual e deve ser mantida em segredo. A chave pública, por sua vez, pode ser compartilhada abertamente ao público. Assim, quando o emissor deseja enviar algo, ele criptografa utilizando sua chave privada, e somente quem tiver a chave pública consegue descriptografar a mensagem. Da mesma forma, o receptor pode criptografar uma mensagem usando a chave pública do emissor, e a única pessoa que teria acesso a essa mensagem descriptografada é o emissor, que utilizaria a sua chave privada para descriptografar.

Na Figura 11, temos a ilustração dos processos de criptografia usando apenas uma chave secreta (a) e utilizando o sistema de chaves assimétricas (b). Notem que as chaves utilizadas são diferentes. No primeiro caso, a mesma chave é usada para os processos de criptografia e decriptografia, enquanto que no segundo caso, uma chave é usada para criptografia, e a outra é usada para descriptografar.

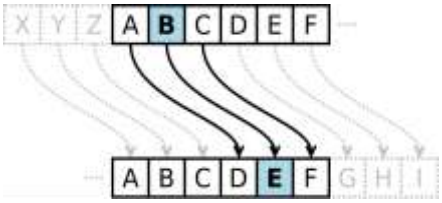
Figura 11. Comparação entre chaves simétricas e assimétricas



Fonte: FOROUZAN, 2009.

Primeiro vamos discutir o conceito de criptografia simétrica, que é algo milenar, utilizado há muito tempo. Talvez você já tenha ouvido falar da "Cifra de César". Foi uma forma de troca de mensagens cifradas em épocas de guerra na Roma antiga, e é uma das mais simples e conhecidas técnicas de criptografia. Ela funciona com o conceito de substituição de letras, onde basicamente definimos um passo de deslocamento entre as letras. Existe um mito que César utilizava o deslocamento de 3 letras.

Figura 12: Cifra de César



Fonte: https://pt.wikipedia.org/wiki/Cifra_de_C%C3%A9sar. Acesso em: 08 jun. 2022.

A cifra de substituição troca um símbolo por outro, e desta forma, a chave que precisamos trocar entre emissor e receptor é o valor de deslocamento. Por exemplo, se tomarmos 3 como passo, o receptor deve ter o conhecimento também, senão ele não conseguirá descriptografar a mensagem. Um exemplo de mensagem com diferentes passos segue abaixo.

Mensagem: OLA

Deslocamento: 3

Mensagem cifrada: ROD

Deslocamento: 7

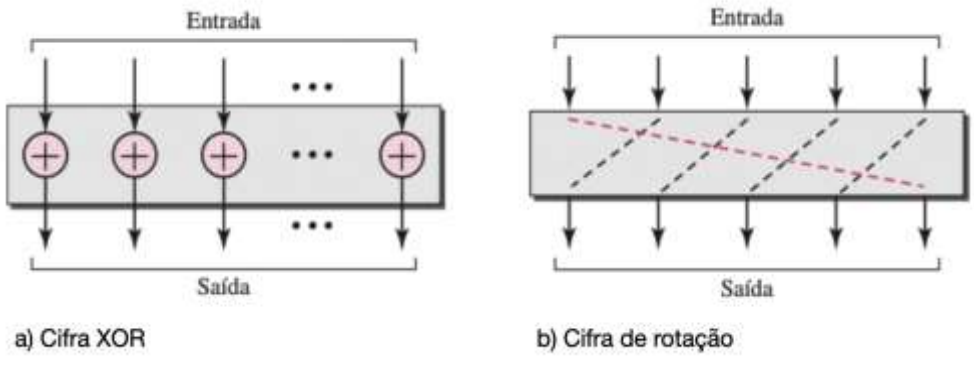
Mensagem cifrada: VSH

Deslocamento: 11

Mensagem cifrada: ZWL

Como sabemos, com o computador, precisamos de cifras que trabalhem com bits, pois estamos lidando com inúmeros dados como texto, áudio, imagem, vídeo e etc. Para converter esses tipos de dados, trabalhamos com fluxos de bits. A ideia principal continua sendo a mesma: rotacionar, deslocar e alterar a informação. As cifras simples, são funções predefinidas simples em matemática e computação. A primeira é a **cifra XOR**, que utiliza a operação OU-EXCLUSIVO. Uma das entradas é o bloco a ser criptografado, a outra entrada é uma chave; o resultado é o bloco criptografado. Outra cifra comum é a **cifra de rotação**, na qual os bits de entrada são deslocados para a esquerda ou para a direita. A cifra de rotação apresenta uma propriedade interessante. Se o comprimento do fluxo original for N, após N rotações obtém-se o fluxo de entrada original. Isso significa que ele é inútil para aplicar mais de N – 1 rotações. Ou seja, o número de rotações deve se situar entre 1 e N – 1.

Figura 13: Cifra XOR e cifra de rotação



Fonte: Adaptado de FOROUZAN, 2009.

Além dessas cifras simples, existem outras que não vamos conseguir cobrir neste material, mas é discutido nos livros de referência. Busquem ler para maiores detalhes das outras cifras simples, pois os próximos algoritmos utilizam as cifras simples em conjunto para formar algoritmos mais robustos.

Videoaula 1

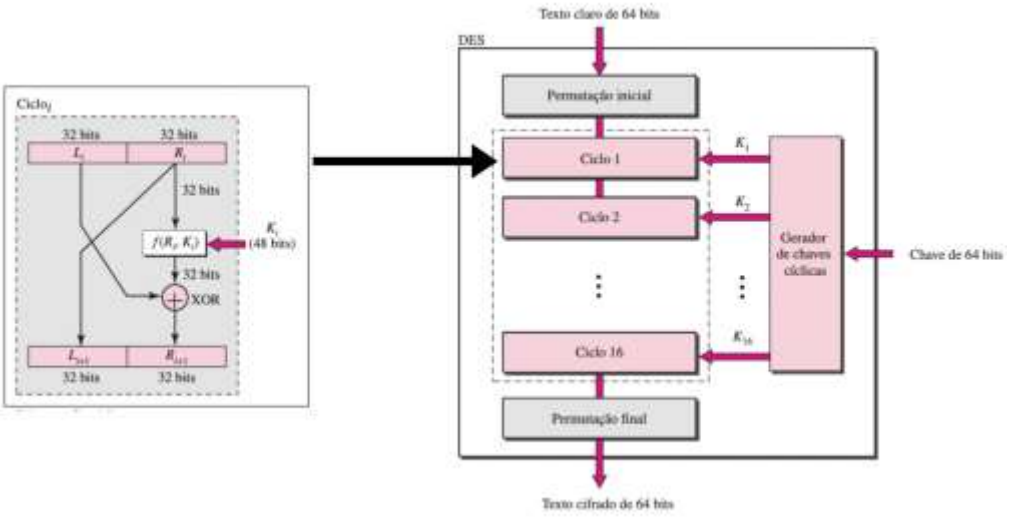
Utilize o QR Code para assistir!

Agora, assista ao vídeo em que o professor aborda o conteúdo inicial de criptografia.

Os algoritmos comercialmente mais utilizados e modernos são: DES e AES. O algoritmo **DES (*Data Encryption Standard*)** é uma cifra de bloco de chave simétrica criada no início da década de 1970 por uma equipe da IBM e adotada pelo *National Institute of Standards and Technology* (NIST). O algoritmo pega o texto simples em blocos de 64 bits e os converte em texto cifrado usando chaves de 48 bits.

O DES possui dois blocos de transposição (P-boxes) e 16 cifras cíclicas complexas (elas são repetidas). Embora as 16 cifras cíclicas de iteração sejam conceitualmente as mesmas, cada uma usa uma chave diferente derivada da chave original. Cada ciclo do DES é uma cifra cíclica complexa. As permutações inicial e final são permutações diretas sem chaves que são invertidas entre si. A permutação usa uma entrada de 64 bits e as permuta de acordo com valores predefinidos.A parte central do DES é a função DES, que aplica uma chave de 48 bits para os 32 bits R_i mais à direita, de modo a gerar uma saída de 32 bits.

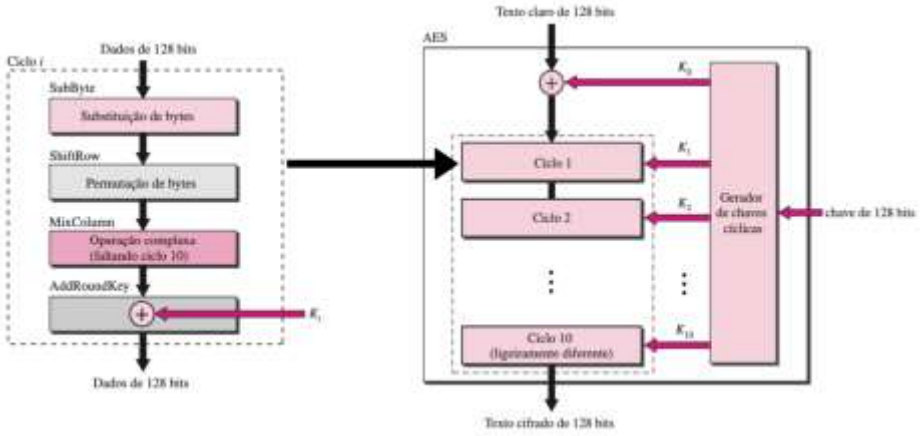
Figura 14: DES e um ciclo DES



Fonte: Adaptado de FOROUZAN, 2009.

O DES recebeu muitas críticas por ter uma chave muito curta, e para aumentar a robustez do algoritmo, foram propostas melhorias como o 3DES duas ou três chaves, com isto, o aumento da chave foi para 112 bits e 168 bits. O **AES (*Advanced Encryption Standard*)** foi projetado para ter três tamanhos de chave: 128, 192 e 256 bits. Há uma operação XOR inicial seguida por dez cifras cíclicas, sendo o último ciclo ligeiramente distinto dos ciclos precedentes e cada bloco usa uma chave diferente derivada da chave original.

Figura 15: AES e um ciclo AES

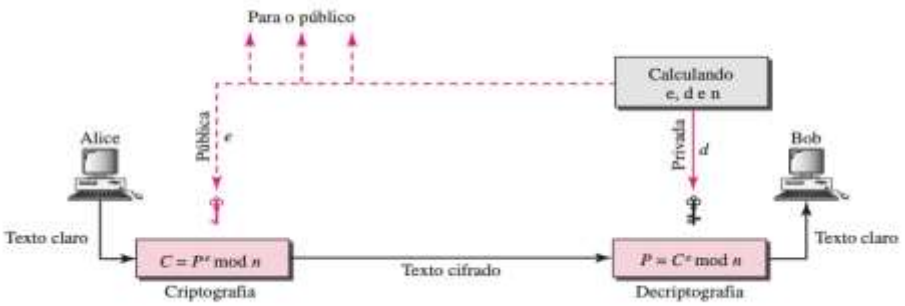


Fonte: Adaptado de FOROUZAN, 2009.

Claro que não existem apenas esses dois algoritmos, existe muita pesquisa na área, mas a maioria dos algoritmos diferem no tamanho do bloco ou chave, número de ciclos e as funções dentro do ciclo. Alguns algoritmos são: IDEA (International Data Encryption Algorithm), Blowfish, CAST-128 e RC5.

Mas esses algoritmos são para criptografia simétrica, do lado dos algoritmos de chave assimétrica, temos: RSA e Diffie-Hellman. O RSA, é assim chamado em homenagem a seus inventores Rivest, Shamir e Adleman (RSA). A ideia simplificada é que ele utiliza dois números, e e d, como chaves pública e privada. Esses valores possuem uma relação especial entre si, são números primos extremamente grandes, mas, o segundo número não pode ser derivado do primeiro. Eis aí um ponto de complexidade elevado.

Figura 15: RSA



Fonte: FOROUZAN, 2009.

Os passos do algoritmo são (retirado de FOROUZAN, 2009):

1. Escolhe dois números primos muito grandes, p e q. Lembre-se de que um número primo é aquele que pode ser dividido igualmente somente por 1 e por si próprio.
2. Multiplica os dois primos escolhidos para descobrir n, o módulo para criptografia e decriptografia. Em outras palavras, $n = p \times q$.
3. Calcula outro número $\phi = (p - 1) \times (q - 1)$
4. Escolhe um inteiro aleatório e. Em seguida, calcula d de modo que d é = $1 \; mod \; \phi$.
5. Anuncia e e n para o público; ele mantém ϕ e secretos

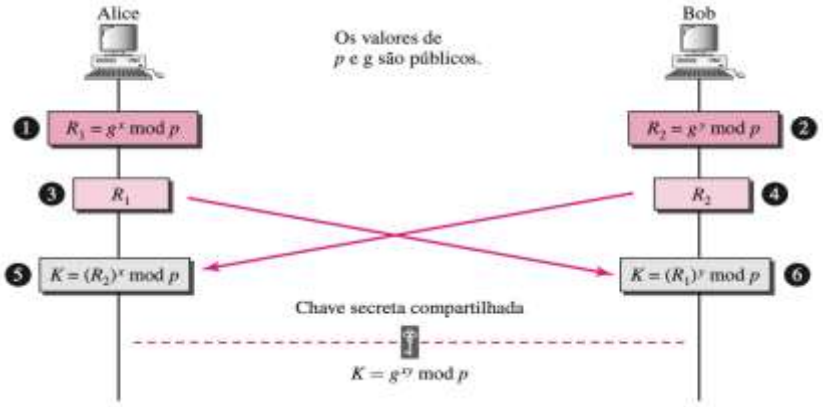
O RSA é muito lento, por isso ele é útil para envio de pequenas mensagens e usado em assinaturas digitais que normalmente precisam criptografar uma pequena mensagem sem ter acesso a uma chave simétrica.

Diffie-Hellman, por sua vez, é um algoritmo desenvolvido para a troca de chaves simétricas. Não é necessário se encontrar para chegar a um acordo de chave, sendo perfeito para usar na Internet. Emissor e receptor criam uma chave de sessão simétrica para troca de dados sem ter de armazenar a chave para uso futuro. Os passos do algoritmo são (FOROUZAN, 2009):

1. Alice escolhe um número aleatório grande x e calcula $R_1 = g^x \; mod \; p$
2. Bob escolhe outro número aleatório grande y e calcula $R_2 = g^y \; mod \; p$
3. Alice envia R_1 para Bob. Note que Alice não envia o valor de x; ela envia apenas R_1.
4. Bob envia R_2 para Alice. Perceba, novamente, que Bob não envia o valor de y, mas apenas R_2 .
5. Alice calcula $K = (R_2)^x \; mod \; p$
6. Bob também calcula $K = (R_1)^y \; mod \; p$

A chave simétrica para a sessão é K. A Figura 16 ilustra esse processo.

Figura 16: Diffie-Hellman



Fonte: FOROUZAN, 2009.

Videoaula 2

Utilize o QR Code para assistir!

Agora, assista ao vídeo em que o professor aborda os algoritmos de criptografia apresentados.

Mas esses algoritmos de criptografia servem para nos fornecer alguns serviços, ou princípios relacionados à segurança: confidencialidade, integridade, autenticação e não-repúdio.

Confidencialidade: Quando um emissor envia uma mensagem para o receptor, é esperado que apenas eles conheçam o conteúdo da mensagem, isso significa ter sigilo sobre a mensagem. A forma de obter isso foi discutida anteriormente, aplicando um algoritmo de criptografia simétrica ou assimétrica, criptografando o conteúdo da mensagem, deixando ela ininteligível para qualquer outra pessoa fora da comunicação.

Integridade: Quando um emissor envia uma mensagem para o receptor, é esperado que o conteúdo da mensagem chegue no emissor da mesma forma quando foi enviado, ou seja, sem nenhuma alteração no conteúdo. A criptografia fornece confidencialidade, mas não integridade. Uma maneira de preservar a integridade de um documento é por meio de uma impressão digital, ou seja, aplicar uma **função *hash*** e gerar o que chamamos de ***digest***. O algoritmo mais utilizado é o SHA-1 (Secure Hash Algorithm 1).

Autenticação: Quando o receptor recebe uma mensagem, ele tem que ter a certeza da identidade do emissor. Uma função hash garante a integridade de uma mensagem, ou seja, que mensagem não tenha sido alterada, mas não garante a autenticidade do emissor. Para isto, temos que fornecer autenticação de mensagens, precisamos de uma forma de assinatura, igual quando assinamos documentos para demonstrar que autenticamos a informação do

documento. Isto pode ser feito através de duas formas: assinando o documento ou assinando um “digest” do documento com assinatura digital.

Não-repúdio: Significa que um emissor não deve ser capaz de rejeitar uma mensagem que ele, de fato, enviou. Isso também pode ser realizado com a assinatura digital, uma vez que é necessário a utilização das chaves pública e privada. De qualquer forma, não é uma forma que solucione o problema totalmente. Uma vez que as partes podem alegar ter trocado as chaves. Para isto, entra na solução, uma terceira parte confiável, uma autoridade de certificação. Essa autoridade teria condições de mostrar uma cópia da mensagem salva.

Videoaula 3

Utilize o QR Code para assistir!

Agora, assista ao vídeo em que o professor aborda todo o conteúdo dos princípios de segurança, e encerro nossa última unidade.

Encerramento

Nesta unidade você aprendeu quais os objetos de estudo da camada de aplicação, e para isto, passamos pelas aplicações mais comuns como correio eletrônico, transferência de arquivos e o http. Aprendemos também sobre o DNS, que é uma aplicação que dá suporte para as outras aplicações funcionarem corretamente. Ainda, vimos um pouco sobre Segurança da Informação, passando pelos elementos mais importantes como os conceitos de criptografia simétrica e assimétrica, e os princípios da segurança. Tentei exemplificar com situações concretas e objetivas. Infelizmente nosso espaço é reduzido e existem muitos outros assuntos dentro de redes de computadores para serem abordados, espero ter instigado você a procurar mais.

Enfim, chegamos ao final, você fechou o ciclo ao conhecer todas as camadas que envolvem redes de computadores. Infelizmente nosso espaço é reduzido e existem muitos outros assuntos dentro de redes de computadores para serem abordados, espero ter instigado você a procurar mais e mais.

Videoaula Encerramento

Utilize o QR Code para assistir!

Assista agora ao vídeo de encerramento de nossa disciplina.

Encerramento da Disciplina

Prezado aluno, agora que você chegou ao fim deste curso, espero que você tenha aproveitado bem o conteúdo dele. Rede de computadores é uma disciplina muito importante, além de estar presente no nosso dia a dia, cada vez mais tem se tornado essencial, pois praticamente tudo que utilizamos hoje está conectado de alguma maneira.

Estudamos os conceitos básicos de rede de computadores, a ideia de trabalhar em camadas e como elas estão organizadas e funcionam integradas para prover serviços para as camadas superiores. Apesar de ser um tópico considerado complexo, espero que eu tenha trazido para vocês uma visão geral, e espero ter despertado o interesse pelo campo também.

Particularmente, acho redes de computadores uma disciplina mágica, o como fazemos as máquinas conversarem entre si para que possamos manter conexão (afetiva) com pessoas mesmo estando a quilômetros de distância, e num simples toque na tela, abrir um aplicativo e falar como se a pessoa estivesse do nosso lado. Essa conexão só é possível graças às redes de computadores. Você agora adquiriu um entendimento básico da disciplina, e agora, pode se aprofundar no campo que mais lhe interessar.

Desejo boa sorte nessa jornada!

Referências

COMER, Douglas E. **Redes de computadores e internet: abrange transmissão** de dados, ligações inter-redes, web e aplicações. 4. ed. Porto Alegre, RS: Bookman, 2008. 632 p. ISBN 978-85-60031-36-8.

FOROUZAN, Behrouz A. **Comunicação de Dados e Redes de Computadores.** 4 ed. AMGH Editora, 2009.

KUROSE, James F.; ROSS, Keith W. **Redes de computadores e a internet:** uma abordagem top-down. 3. ed. São Paulo: Addison-Wesley, 2006. 634 p.

TANENBAUM, Andrew S.; WHETERALL, David J. **Redes de Computadores.** Pearson Prentice Hall, 2011. p. 582. ISBN 9788576059240.


UniFil
EDUCAÇÃO A DISTÂNCIA
UNIFIL.BR