Proteção de dados em segurança de redes de computadores

Conceitos e Aplicações

Prof. Ronierison Maciel (Roni)

22/11/2023

O que veremos hoje?

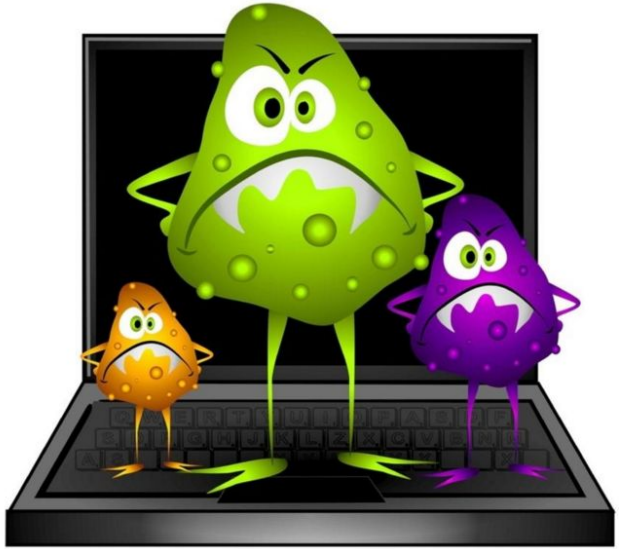
1. Conceitos introdutórios sobre segurança de redes
2. Discussão em grupo sobre notícias recentes de violações de segurança
3. Quizz

Introdução

- Dúvidas 🤔
  - Qual a quantidade de informações produzida a cada dia - 2,5 quintilhões
  - O sistema da minha empresa é seguro?
  - Será que isso acontecerá na minha empresa?
  - Como minimizar os riscos?

Introdução

➔ E quanto às suas informações, você se preocupa com a sua segurança?

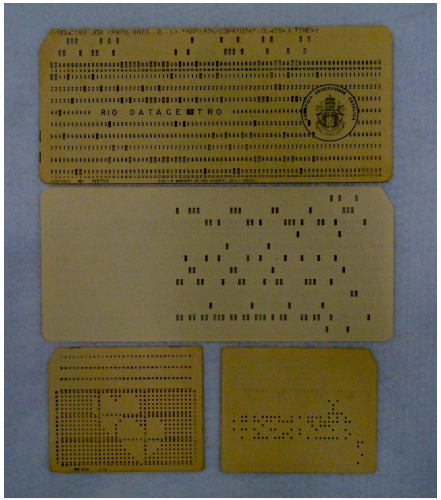


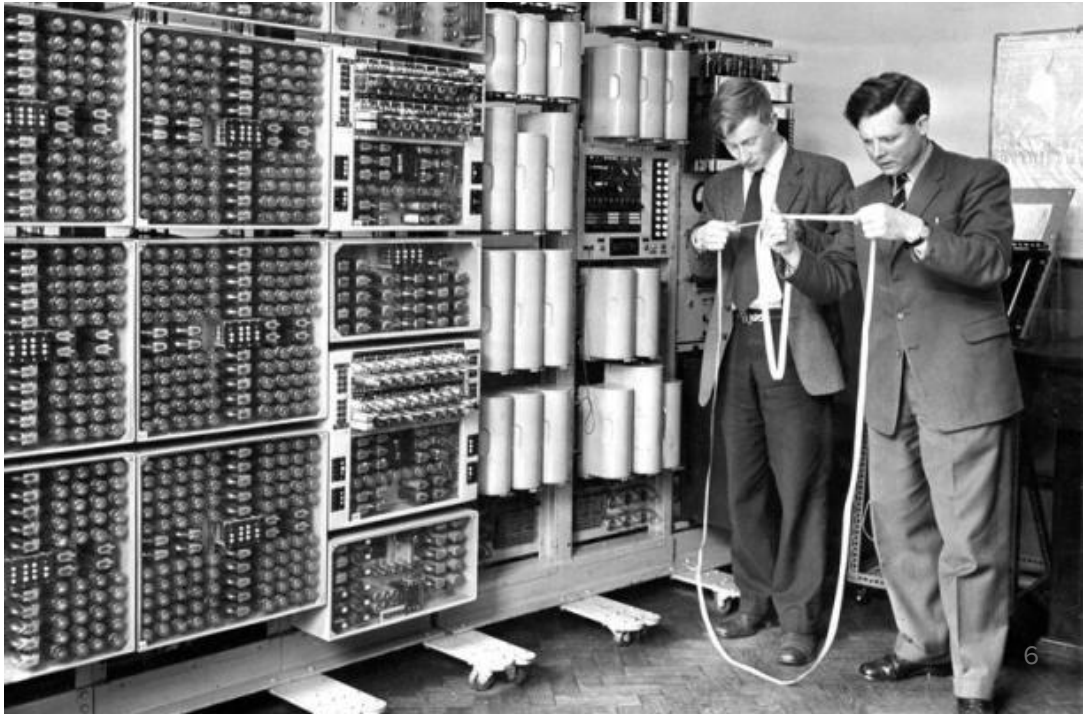
Introdução

➔ O que torna um sistema seguro? 🧐

Introdução

➔ Antigamente





Introdução

➔ Atualmente

Introdução

➔ Ameaças

Introdução

- Informação
  - “O bem mais valioso”

Objetivos do Curso e Metodologia

➔ **Ponto 1:**

- Compreender os conceitos fundamentais da segurança de redes.

Ponto 1

➔ **Fatores secundários:**

- Intempéries da natureza;
- Greves;
- Manifestações.

Objetivos do Curso e Metodologia

➔ **Ponto 2:**

- Reconhecer a importância da segurança física e lógica.

Ponto 2

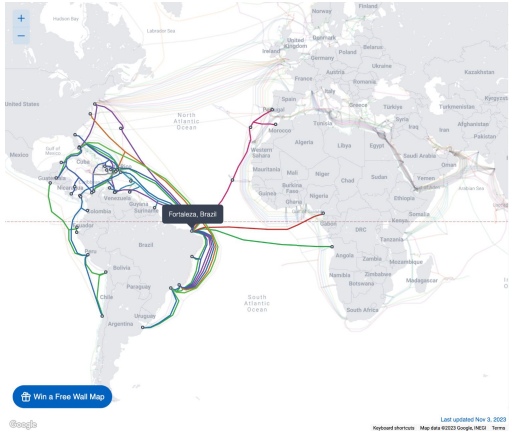
➔ Ameaças

- Um agente externo que oferece perigo.

Objetivos do Curso e Metodologia

➔ **Ponto 3:**

- Casos reais e recentes de violações de segurança 🔥
- Cable Map
- 3D Map



Ponto 3

➔ Vulnerabilidades

- É uma fraqueza que permite que um atacante reduza a garantia da informação de um sistema.

Ponto 3+

- **Vulnerabilidades**
  - Físicas;
  - Naturais;
  - Hardware;
  - Software;
  - Mídias;
  - Comunicação;
  - Humanas.

Fluxograma da vulnerabilidade

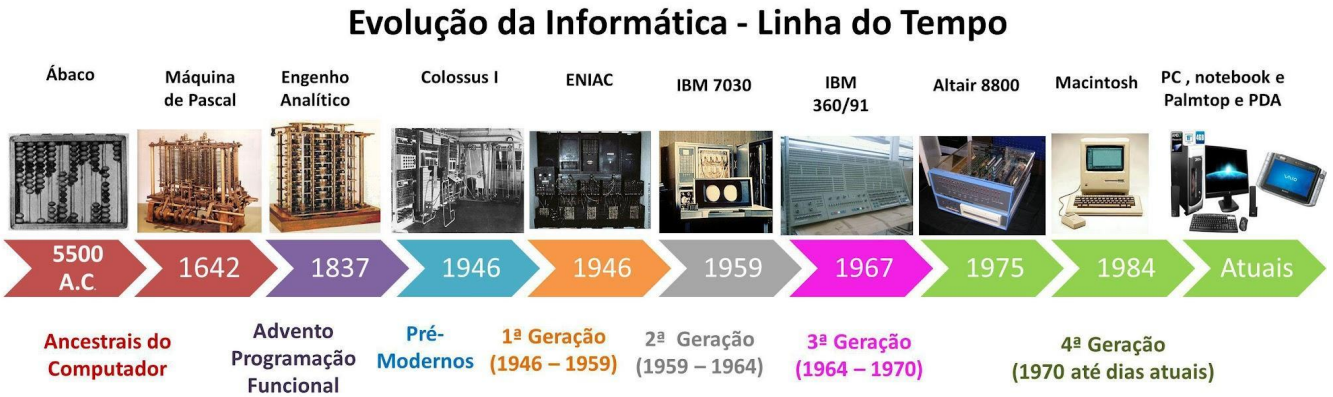
Objetivos do Curso e Metodologia

- Quizz
  - Quizizz.com

Uma Viagem no Tempo

➔ A evolução da segurança Digital

- Início dos anos **1970**: Primeiros vírus em mainframes.
- Anos **1980**: Surgimento dos primeiros antivírus.
- Anos **1990**: A era da Internet e novos desafios de segurança.
- Anos **2000**: Ataques cibernéticos de larga escala e o início da guerra cibernética.
- Anos **2010**: Aumento de ataques sofisticados e o papel da IA na segurança.



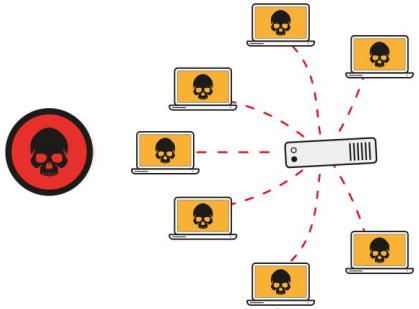
O Mundo Digital Hoje: Números que Falam

- Custo das Violências:
  - Custo médio de uma violação de dados: US$ 4,35 milhões (2022).
  - Custo médio de um ataque de ransomware: US$ 4,54 milhões.
  - Custo médio de recuperação de ransomware: quase US$ 2 milhões.
  - Nos **EUA**, o custo médio de violação de dados é de US$ 9,44 milhões.
  - O custo médio de uma violação de dados no **Brasil** foi de R$ 7,71 milhões [1].
- Frequência e Duração:
  - Empresas levaram em média 277 dias para identificar e conter uma violação (2022).
  - A cada minuto, US$ 17.700 são perdidos devido a ataques de phishing.
  - Em média, um ataque cibernético ocorre a cada três segundos.
  - No Brasil, as empresas levaram em média 347 dias para identificar e conter uma violação de dados em 2022, o que representa uma redução de 49 dias em relação ao ano anterior.[2]

Introdução aos Tipos de Ataques

- **Ataques de negação de serviço (DoS/DDoS):**
  - Esses ataques visam impedir que um sistema ou rede funcione. Eles são geralmente realizados enviando um grande número de solicitações a um servidor, o que pode sobrecarregá-lo e torná-lo inacessível.
- **Phishing:**
  - Esses ataques visam enganar as pessoas para que divulguem informações confidenciais, como senhas ou números de cartão de crédito. Eles geralmente são realizados por meio de e-mails ou mensagens de texto que parecem ser legítimos.
- **Ransomware:**
  - Esses ataques criptografam os dados da vítima e exigem um resgate para descriptografá-los. Eles geralmente são distribuídos por meio de anexos de e-mail ou downloads infectados.

Ataque DDoS/DoS

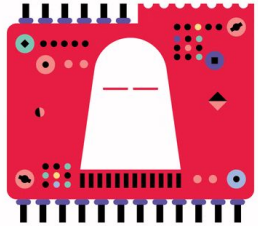


- **Inundação de Rede:**
  - Onde o atacante sobrecarrega a capacidade de rede do alvo com um alto volume de tráfego.
- **Exploitation de Vulnerabilidades de Software:**
  - Atacando as fraquezas específicas de um software para causar falhas ou sobrecarga.
- **Ataques ao Nível de Aplicação:**
  - Especificamente direcionados para aplicativos web, enviando um número excessivo de pedidos que parecem legítimos mas têm como objetivo drenar os recursos do servidor.
- **Casos reais:**
  - O ataque de 2000 contra o site da Amazon, que deixou o site inacessível por várias horas.
  - O ataque de 2016 contra o site da Sony Pictures, que deixou o site inacessível por vários dias.
  - O ataque de 2020 contra o site do governo dos Estados Unidos, que deixou o site inacessível por várias horas.

Ataque Phishing

- **E-mails de Phishing:**
  - Imagine um personagem recebendo um e-mail que parece ser de seu banco, pedindo que atualize suas informações de segurança. O e-mail contém um link que leva a um site falso, indistinguível do real, onde as informações inseridas são roubadas pelo atacante.
- **Spear Phishing:**
  - Mais direcionado, como um caçador que mira em uma presa específica. Aqui, o atacante conhece informações sobre a vítima e personaliza o ataque para torná-lo mais convincente. Por exemplo, um e-mail que parece vir de um colega de trabalho com um pedido para verificar um documento.
- **Phishing via SMS (Smishing):**
  - Mensagens de texto fraudulentas que levam as vítimas a sites maliciosos ou as induzem a divulgar informações pessoais. Pode ser representado como uma mensagem urgente pedindo ação imediata.
- **Phishing por Voz (Vishing):**
  - Envolve telefonemas, onde o atacante se faz passar por uma autoridade ou empresa confiável para extrair informações pessoais. Imagine uma cena em que um personagem recebe uma ligação de seu "banco" informando sobre uma atividade suspeita em sua conta.

Ataque Ransomware



Ransomware é um tipo de malware (software malicioso) que criptografa os arquivos do computador da vítima, tornando-os inacessíveis. O atacante então exige um pagamento (geralmente em criptomoeda, para manter o anonimato) para fornecer a chave que pode descriptografar os arquivos. É como um sequestro digital, onde os dados são os reféns.

- **Infecção:**
  - O ransomware geralmente se infiltra em um sistema através de e-mails de phishing, downloads maliciosos ou exploração de vulnerabilidades de segurança. Pode-se imaginar isso como um invasor sorrateiro que encontra uma janela destrancada para entrar em uma casa.
- **Criptografia:**
  - Uma vez dentro do sistema, o ransomware criptografa arquivos importantes. Esta etapa é como colocar cadeados em objetos de valor, onde apenas o ladrão tem a chave.
- **Exigência de Resgate:**
  - O atacante então exibe uma mensagem exigindo pagamento para desbloquear os arquivos. É como um sequestrador enviando uma nota de resgate, com instruções específicas e um prazo.

Casos Notórios

- **WannaCry (2017):**
  - Um dos ataques de ransomware mais famosos, afetou centenas de milhares de computadores em mais de 150 países. Explorou uma vulnerabilidade no Windows, afetando principalmente hospitais, empresas e instituições governamentais.
- **NotPetya (2017):**
  - Originalmente direcionado à Ucrânia, espalhou-se globalmente, causando bilhões de dólares em danos. Foi mais destrutivo que lucrativo, pois muitas vezes os dados não podiam ser recuperados mesmo após o pagamento.
- **Colonial Pipeline (2021):**
  - Um dos maiores ataques de ransomware nos EUA, afetou o fornecimento de combustível na Costa Leste. A empresa pagou um resgate significativo em Bitcoin, destacando a vulnerabilidade da infraestrutura crítica.

Data Center Locaweb

Data Center Google

Segurança Data Center da Google

Bem-vindos à Semana 2 do nosso curso!

Esta **semana**, mergulharemos profundamente no mundo dos ataques cibernéticos, explorando em **detalhes** as diversas formas que esses **ataques** podem assumir. Nossa jornada nos levará a **compreender** não apenas o que esses **ataques** são, mas também como eles **funcionam**, como **identificá-los** e, o mais importante, como nos proteger contra eles.

Objetivos da Semana

- **Entender e Diferenciar Tipos de Ataques:**
  - Vamos explorar os detalhes de ataques como Backdoor, DoS, DDoS, DMA, Eavesdropping e Spoofing. Essa compreensão é fundamental para qualquer profissional de segurança de redes.
- **Análise de Casos Reais:**
  - Através da análise de estudos de caso reais, entenderemos o impacto desses ataques no mundo real, agregando uma camada prática ao nosso conhecimento teórico.
- **Desenvolvimento de Habilidades Práticas:**
  - Com exercícios e laboratórios práticos, você terá a oportunidade de aplicar o conhecimento adquirido em situações simuladas, fortalecendo suas habilidades de identificação e mitigação de ataques.

Tópicos a serem Abordados

- **Backdoor:**
  - Exploraremos o que são Backdoors, como eles são implantados e como podem ser detectados e prevenidos.
- **DoS e DDoS:**
  - Entenderemos a diferença entre ataques DoS e DDoS, seus impactos e estratégias de defesa.
- **DMA Attack:**
  - Discutiremos ataques via Direct Memory Access e como se proteger contra eles.
- **Eavesdropping:**
  - Analisaremos técnicas de escuta indesejada e como garantir a confidencialidade dos dados.
- **Spoofing:**
  - Estudaremos diferentes tipos de Spoofing e como identificar e proteger redes contra essas ameaças.

NMAP

- Principais comandos com NMAP:
  - nmap -p- [endereço IP ou nome do host]
    - Realiza uma varredura padrão nas 1.000 portas TCP mais comuns.
  - nmap -sV [endereço IP]
    - Detecta versões de serviços em execução nas portas abertas.
  - nmap -O [endereço IP]
    - Tenta identificar o sistema operacional do alvo.
  - nmap -T4 -A [endereço IP]
    - Realiza uma varredura agressiva (mais rápida e com detecção de versões de serviços, sistema operacional, traceroute, e scripts padrão).
  - nmap -sU [endereço IP]
    - Escaneia portas UDP (útil para encontrar serviços que não estão escutando em portas TCP padrão).

Recursos

- Ferramentas:
  - Slowloris
  - LOIC
  - Wireshark
  - Collection tools
- Plataformas
  - Hack The Box
  - TryHackMe
  - Hacker One
- Ambiente
  - VulnHub

Proteção de Dados e Criptografia

Criptografia e Técnicas de Codificação

➔ Introdução à criptografia.

- A criptografia é um método essencial para proteger informações em redes de computadores.
- Stallings explica que a criptografia transforma dados originais **(texto claro)** em uma forma ininteligível **(texto cifrado)** usando um algoritmo e uma chave.
- Ela serve para garantir a confidencialidade dos dados, assegurando que somente quem possui a chave correta possa decifrá-los.

Criptografia e Técnicas de Codificação

➔ Criptografia simétrica vs. assimétrica.

- **Criptografia Simétrica:** Aqui, a mesma chave é utilizada tanto para cifrar quanto para decifrar os dados. Stallings destaca sua eficiência em termos de velocidade, mas aponta o desafio de compartilhar a chave de forma segura.
- **Criptografia Assimétrica:** Utiliza um par de chaves – uma pública e outra privada. A chave pública pode ser compartilhada livremente para cifrar uma mensagem, mas somente a chave privada correspondente pode decifrá-la. Stallings ressalta a segurança aprimorada deste método, mas nota que ele é mais lento comparado à criptografia simétrica.

Criptografia e Técnicas de Codificação

➔ Chave pública vs. chave privada

- **Chave Pública:** Faz parte do sistema de criptografia de chave pública, onde cada usuário possui uma chave pública, que pode ser compartilhada abertamente. Essa chave é usada para criptografar mensagens ou verificar assinaturas digitais.
- **Chave Privada:** Também conhecida como chave secreta, é mantida em segredo pelo usuário e é usada para descriptografar mensagens criptografadas com a chave pública correspondente ou para assinar digitalmente um documento.

Criptografia e Técnicas de Codificação

- **Exemplos de algoritmos: AES, RSA, DES.**
  - **AES (Advanced Encryption Standard):** É um exemplo de criptografia simétrica. Stallings explica que o AES é amplamente usado devido à sua força e eficiência, sendo uma escolha padrão em muitas aplicações de segurança.
  - **RSA (Rivest-Shamir-Adleman):** Este é um exemplo de criptografia assimétrica. No livro, o RSA é descrito como um dos primeiros e mais utilizados algoritmos de criptografia de chave pública, sendo fundamental para muitos protocolos de segurança na internet.
  - **DES (Data Encryption Standard):** É um algoritmo de criptografia simétrica, onde a mesma chave é usada para criptografar e descriptografar a informação. O DES foi amplamente utilizado, mas é considerado obsoleto devido à sua chave relativamente curta de 56 bits, que o torna vulnerável a ataques de força bruta.

Criptografia e Técnicas de Codificação

➔ **Exemplos de algoritmos: RC4, Rabbit e 3DES.**

- **RC4:** É um algoritmo de criptografia de fluxo. Ele gera um fluxo de bits pseudo aleatórios que são combinados com o texto claro (plaintext) usando uma operação de ou-exclusivo (XOR). RC4 foi popular em protocolos como WEP e TLS, mas foi substituído devido a vulnerabilidades.
- **Rabbit:** É um algoritmo de criptografia de fluxo desenvolvido para alta performance. Ele é conhecido por sua velocidade e segurança, mas não é tão amplamente utilizado quanto outros algoritmos de criptografia de fluxo.
- **TripleDES (3DES):** É uma evolução do DES que aplica o algoritmo DES três vezes a cada bloco de dados. O 3DES foi projetado para oferecer uma segurança mais robusta através do aumento do tamanho da chave, embora com uma redução na velocidade de processamento.

Exemplos de Aplicações da Criptografia

- **Transações Financeiras Online:**
  - Bancos e serviços de pagamento online utiliza criptografía SSL/TLS para proteger as transações financeiras. Isso garante que informações sensíveis, como números de cartão de crédito, sejam transmitidas de forma segura.
- **Comunicação Empresarial:**
  - Empresas utilizam plataformas de comunicação seguras, como e-mails e mensageiros criptografados (ex.: Signal, WhatsApp), para proteger a troca de informações sensíveis.
- **Armazenamento em Nuvem:**
  - Dados armazenados em serviços de nuvem, como AWS ou Google Cloud, são frequentemente criptografados para proteger contra acessos não autorizados.

Impactos da Ausência de Criptografia

- **Vazamento de Dados Sensíveis:**
  - Sem criptografia, informações como dados pessoais e corporativos podem ser interceptadas e acessadas por atores mal-intencionados.
- **Fraude e Roubo de Identidade:**
  - Informações financeiras desprotegidas podem levar a fraudes e roubos de identidade, causando perdas financeiras significativas.
- **Perda de Confiança do Cliente:**
  - A ausência de segurança adequada pode levar à perda da confiança do cliente e danos à reputação da empresa.

Empresas danos por falhas na Criptografia

- **Vazamento de Dados Pessoais em 2021:**
  - Foi descoberto em janeiro de **2021** o maior vazamento de dados pessoais na história do Brasil, expondo informações de **223 milhões de brasileiros**. Os dados incluíam nomes, identificadores fiscais únicos, imagens faciais, endereços, números de telefone, e-mails, pontuação de crédito, salários e mais. Há suspeitas de que o vazamento tenha origem na Serasa Experian, uma importante agência de crédito no Brasil, embora a empresa negue. Este incidente é notável por seu tamanho e por incluir informações até de indivíduos falecidos.
- **Ataques a Grandes Empresas:**
  - Empresas como **Porto Seguro**, uma das maiores seguradoras do Brasil, e a **CVC**, uma das maiores agências de viagens do país, sofreram ataques cibernéticos recentemente. Esses casos destacam a luta contínua contra ameaças cibernéticas..
- **Aumento de Ataques Cibernéticos Durante a Pandemia:**
  - Desde o início da pandemia de **COVID-19**, houve um aumento dramático no número de ataques cibernéticos a empresas e indivíduos no Brasil. Esse aumento também levou a um crescimento nos gastos com segurança cibernética e a uma maior responsabilidade por incidentes de violação de dados.

Exemplos

Hash

Encrypt

Quiz

Visão Geral de SSL/TLS

➔ O que são SSL e TLS?:

- **SSL** e **TLS** são **protocolos criptográficos** projetados para fornecer comunicações seguras em uma **rede de computadores**. Eles são mais comumente **usados** na **Internet** para **garantir** transações **seguras** e **comunicação** entre navegadores da **web** e **servidores**.

➔ Funcionamento:

- SSL/TLS operam entre as camadas de transporte e aplicação do modelo OSI. Eles utilizam a criptografia para assegurar a confidencialidade e integridade dos dados, além de autenticar a identidade das partes envolvidas na comunicação.

➔ Evolução do SSL para TLS:

- O **TLS** é uma **evolução** do **SSL**, oferecendo **melhorias** em termos de **segurança** e **eficiência**. Enquanto o **SSL** foi desenvolvido pela **Netscape**, o TLS foi padronizado pela Internet Engineering Task Force (**IETF**).

Introdução às VPNs e IPSec

- **VPNs (Virtual Private Networks):**
  - As VPNs são **redes** que permitem a criação de **conexões seguras** e **criptografadas** sobre uma **rede menos segura**, como a Internet. Elas são usadas para **garantir** a **privacidade** e **segurança** dos dados transmitidos.
- **IPSec (Internet Protocol Security):**
  - IPSec é um conjunto de protocolos usados para garantir a segurança das comunicações na camada de rede. Ele é frequentemente usado em VPNs para autenticar e criptografar pacotes de dados enviados através da Internet.
- **Relação com SSL/TLS:**
  - Enquanto SSL/TLS são usados principalmente para segurança na camada de aplicação, especialmente em comunicações web, IPSec opera na camada de rede, fornecendo um escopo mais amplo de proteção e é ideal para VPNs.

Dica!

O **SSL** e **TLS** são como **envelopes** super seguros para proteger suas **informações** na **internet**, e **VPNs** com **IPSec** são como **túneis secretos** e super seguros para enviar informações importantes sem que ninguém veja.

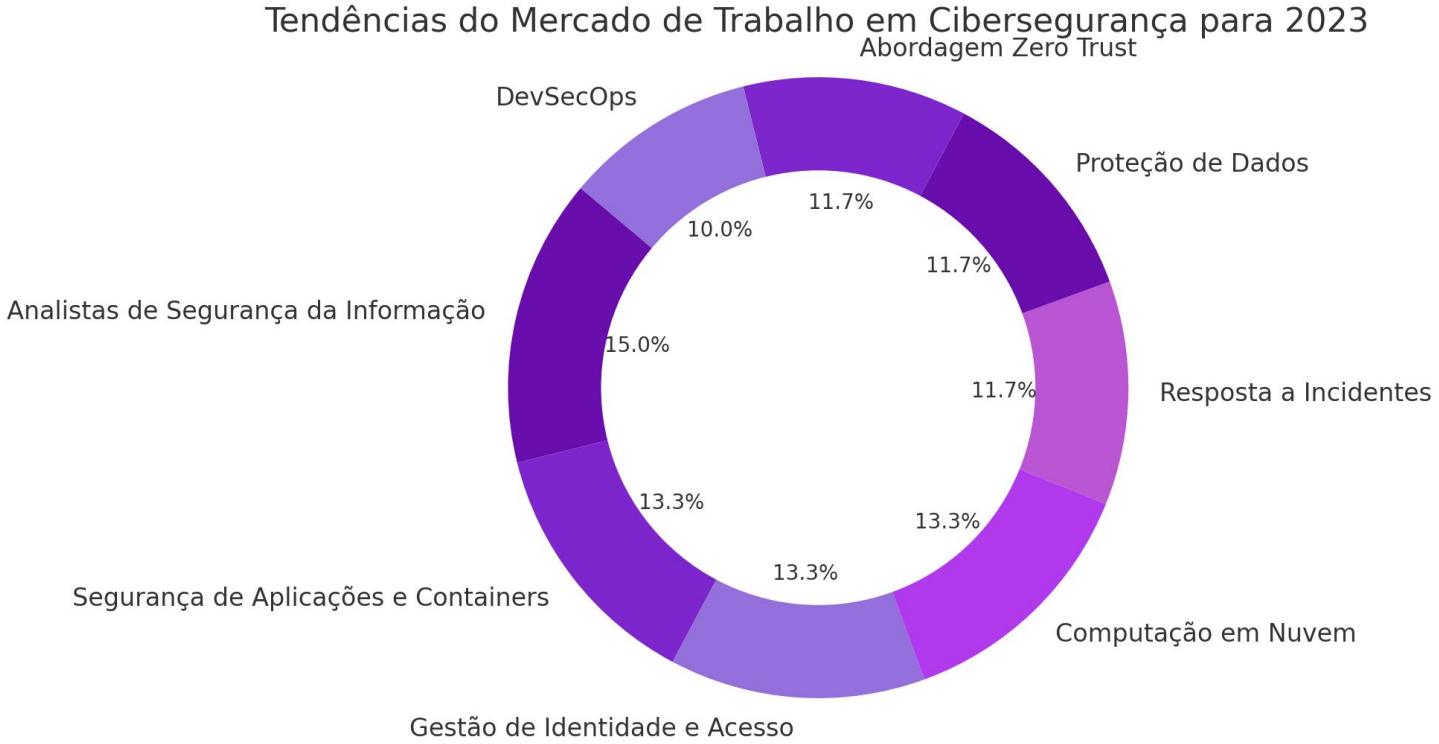
Ferramentas para Testar SSL/TLS

- **SSL Labs' SSL Test:**
  - Uma ferramenta online gratuita que avalia a configuração de segurança de servidores web SSL/TLS.
- **TestSSL.sh:**
  - Um script de linha de comando que realiza uma variedade de verificações em um servidor web SSL/TLS.
- **OpenSSL:**
  - Uma ferramenta poderosa que pode ser usada para testar e configurar servidores SSL/TLS.
- **Qualys SSL/TLS Scanner:**
  - Oferece uma análise detalhada da configuração de SSL/TLS de um servidor.
- **SSLyze:**
  - Uma ferramenta Python para analisar a configuração de SSL/TLS de um servidor, permitindo a automação de testes.
- **Nmap com NSE Scripts:**
  - Nmap, junto com seus scripts de Network Security Evaluation (NSE), pode ser usado para detectar e testar vulnerabilidades em SSL/TLS.

Ferramentas para Testar VPN e IPSec

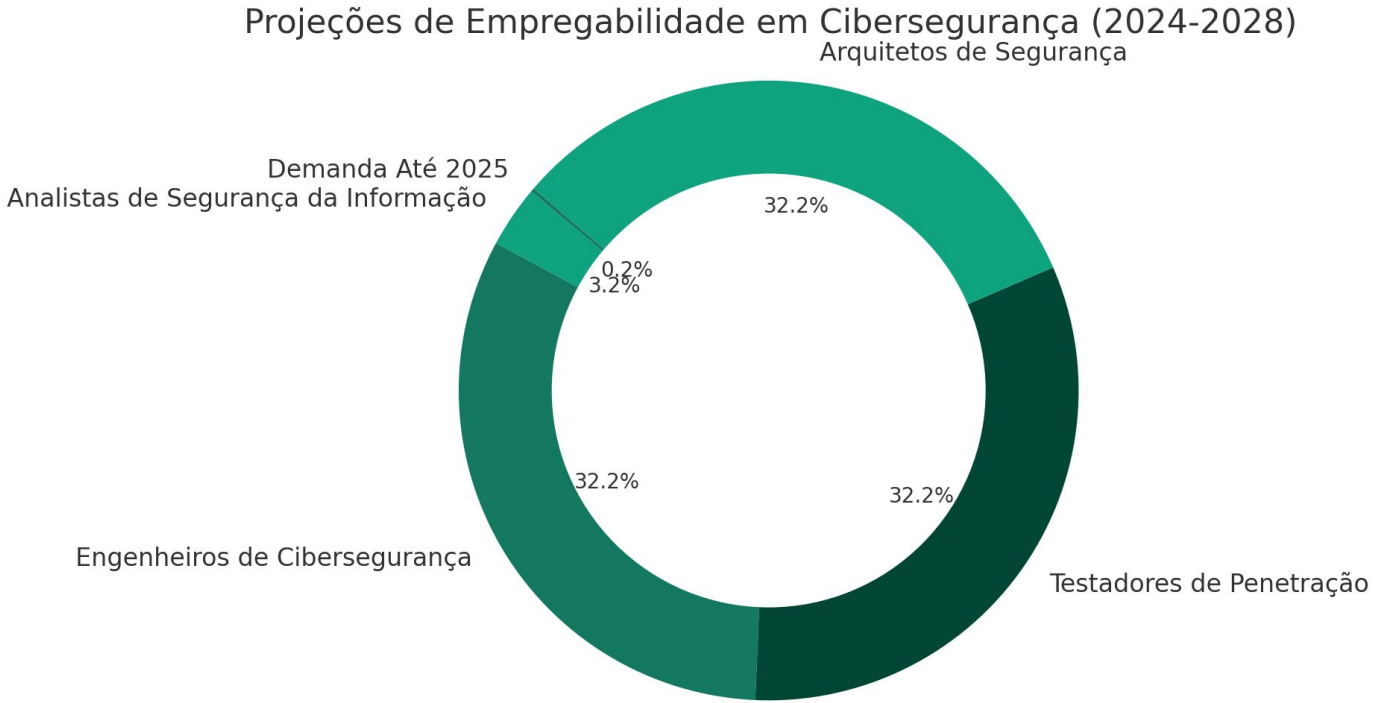
- **Wireshark:**
  - Uma ferramenta de análise de rede que pode capturar e analisar o tráfego IPSec para verificar se os dados estão sendo criptografados corretamente.
- **IKE-scan:**
  - Uma ferramenta específica para testar VPNs que usam o protocolo IPSec, focada em descobrir e identificar gateways IPSec.
- **OpenVPN Test:**
  - Para VPNs que utilizam OpenVPN, esta ferramenta pode testar a configuração e a segurança da implementação.
- **StrongSwan:**
  - Embora seja mais conhecido como uma implementação de IPSec, StrongSwan também possui funcionalidades para testar e verificar conexões IPSec.
- **Nmap:**
  - Assim como para SSL/TLS, Nmap pode ser usado para identificar e testar a segurança de VPNs, incluindo a detecção de portas e serviços associados ao IPSec.
- **tcpdump:**
  - Útil para capturar e analisar o tráfego de rede, incluindo o tráfego IPSec, para verificar se a VPN está funcionando como esperado.

2023



Fonte: Bureau of Labor Statistics, ISACA, Dice.com

2024-2028



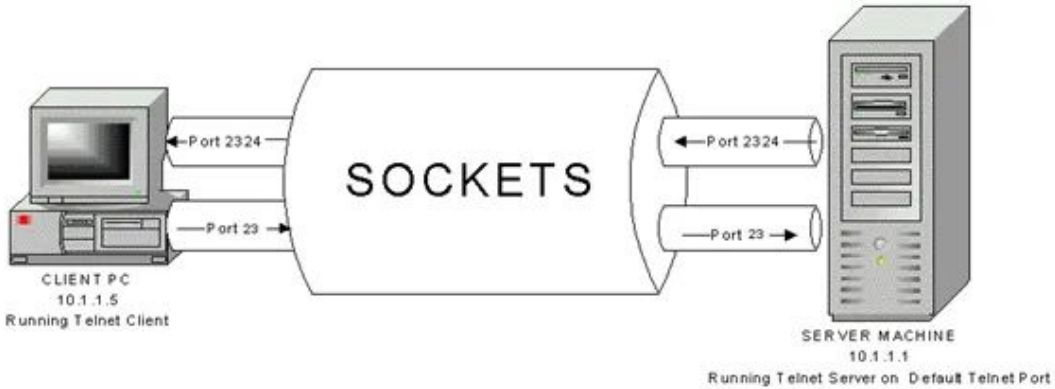
Fonte: Bureau of Labor Statistics, ISACA, Cybersecurity Ventures

Socket - Python

Sockets são ferramentas que possibilitam a **interação** entre **dois** processos (**aplicativos**) distintos, seja no mesmo dispositivo ou em equipamentos separados. Isso implica que os Sockets são empregados para **facilitar** a comunicação entre diversos **computadores** ou entre dois **aplicativos** (processos) que operam simultaneamente no mesmo dispositivo.

Socket - Python

A imagem ilustra a **comunicação** via **socket** entre **dois computadores** diferentes. Neste processo, o computador cliente inicia a comunicação, enquanto o servidor aguarda por solicitações do cliente. Cada comunicação utiliza uma "**Porta**", um número que permite múltiplas comunicações em cada computador sem interferências entre elas. No exemplo, o Cliente usa o Socket na Porta **23** para se comunicar com o Servidor, que responde através de um Socket na Porta 2324.

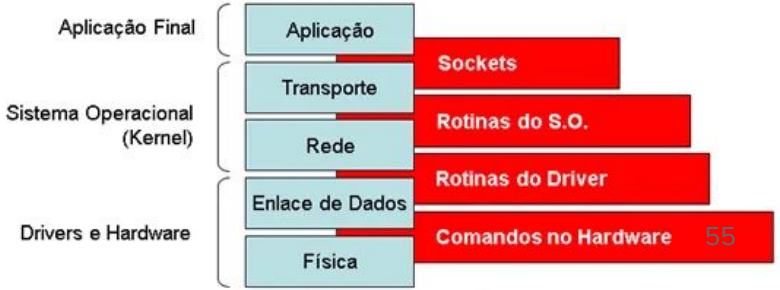


Socket - Python

Quando você digita um endereço web (www.google.com.br) em seu navegador, ele cria um Socket. Neste cenário, você atua como Cliente e o computador onde a página está armazenada é o Servidor. Durante este processo, várias etapas ocorrem internamente nos sistemas operacionais (Windows, Linux, Mac) tanto do Servidor quanto do Cliente, embora o usuário final (você) não perceba.

Socket - Python

A figura demonstra as etapas da comunicação via Socket em cada computador. No lado do Cliente, o navegador funciona como uma interface gráfica que interage com a camada de **Aplicação** do sistema operacional, solicitando a criação de um Socket. Cada camada do sistema (**Aplicação**, **Sistema Operacional**, **Drivers/Hardware**) executa um serviço específico, configura-o e o transmite para a próxima camada. Essas camadas são classificadas de acordo com sua função: **Aplicação Final** (**Navegador**), **Sistema Operacional (Windows, Linux, Mac)** e **Drivers/Hardware** (componentes físicos do computador, como a placa de rede).



Vamos colocar a mão na massa!

SocketPython

A Aventura de uma mensagem pela Internet

Imagine que estamos embarcando em uma jornada pelo vasto mundo da Internet. Nossa missão é enviar uma mensagem até o grande servidor do **Google** e trazer de volta uma **resposta**. Para isso, vamos usar uma ferramenta poderosa: o **socket**.

- ➔ **Importação e Definição do Alvo:**
  - Primeiro, precisamos das ferramentas certas. Aqui, importamos o módulo **socket**, que é como uma mochila cheia de ferramentas para se comunicar pela Internet.
  - Nosso destino? O famoso **www.google.com**, na porta **80**, a porta padrão para a comunicação web não segura.
- ➔ **Criando nosso Veículo de Comunicação (Socket):**
  - Agora, construímos nosso veículo, o socket. Pense nele como um pequeno drone que pode voar pela Internet.
  - Configuramos para usar **IPv4 (AF_INET)** e para ser um veículo confiável e de conexão contínua **(SOCK_STREAM, ou seja, TCP)**.

Continuação

- **Partindo para o Google:**
  - Com nosso drone pronto, conectamos ao Google. Aqui, **client.connect** é como dizer ao drone para voar até o servidor do Google.
- **Enviando a Mensagem:**
  - Precisamos de uma mensagem para enviar. Usamos um formato especial, chamado **HTTP**, que o Google entende. Aqui, estamos pedindo a página principal do Google **(GET / HTTP/1.1)**.
  - Enviamos a mensagem. Nosso drone agora está levando nossa solicitação pelo vasto oceano da Internet.
- **Aguardando a Resposta:**
  - Após o envio, esperamos. O drone coleta a resposta do Google e a traz de volta para nós.
- **Revelando a Mensagem do Google:**
  - Finalmente, imprimimos a resposta. O que será que o Google nos enviou de volta?

Resposta do Google!

➔ **Cabeçalhos HTTP:**

- **Date**, **Expires**, **Cache-Control:** Informam sobre a data da resposta, a data de expiração e as diretrizes de cache.
- **Content-Type:** text/html; charset=ISO-8859-1: Especifica que o tipo de conteúdo da resposta é HTML e o conjunto de caracteres é ISO-8859-1.
- **Content-Security-Policy-Report-Only:** Define a política de segurança de conteúdo do site.
- Server: gws: Indica que o servidor é um servidor Google Web Server (gws).
- **Set-Cookie:** Esses cabeçalhos definem cookies que podem ser usados para manter o estado da sessão ou outras informações.
- **X-XSS-Protection, X-Frame-Options:** Cabeçalhos relacionados à segurança.

Resposta do Google!

- **Status da Resposta: HTTP/1.1 200 OK**
  - **HTTP/1.1** indica que a resposta está usando a versão 1.1 do protocolo HTTP.
  - **200 OK** é um código de status HTTP que indica que a solicitação foi bem-sucedida e o servidor transmitiu a resposta solicitada.
- **Corpo da Resposta:**
  - O corpo da resposta contém o HTML da página inicial do Google. Inclui várias tags HTML (**<html>**, **<head>**, **<body>**, etc.), metadados (como **<meta>** e **<title>**), e scripts JavaScript.
  - A parte inicial do HTML inclui informações sobre o layout da página, estilo CSS, e meta tags para SEO e configurações de mídia social.

Rememorando Criptografia

➔ Introdução à criptografia.

- A criptografia é um método essencial para proteger informações em redes de computadores.
- Stallings explica que a criptografia transforma dados originais **(texto claro)** em uma forma ininteligível **(texto cifrado)** usando um algoritmo e uma chave.
- Ela serve para garantir a confidencialidade dos dados, assegurando que somente quem possui a chave correta possa decifrá-los.

Rememorando Criptografia

- **Criptografia simétrica vs. assimétrica.**
  - **Criptografia Simétrica:** Aqui, a **mesma** chave é utilizada tanto para **cifrar** quanto para **decifrar** os **dados**. Stallings destaca sua eficiência em termos de velocidade, mas aponta o desafio de compartilhar a chave de forma segura.
  - **Criptografia Assimétrica:** Utiliza um **par** de chaves – uma **pública** e outra **privada**. A chave pública pode ser compartilhada livremente para cifrar uma mensagem, mas somente a chave privada correspondente pode decifrá-la. Stallings ressalta a segurança aprimorada deste método, mas nota que ele é mais lento comparado à criptografia simétrica.

O que é Fernet

- ➔ Fernet é uma implementação de criptografia simétrica. Pense nela como um tipo específico de fechadura e chave para o seu cofre de dados. Fernet é seguro e fácil de usar, tornando-o uma escolha popular.
- ➔ Fernet Documentation

Praticando!!

Cloud Google