Q1

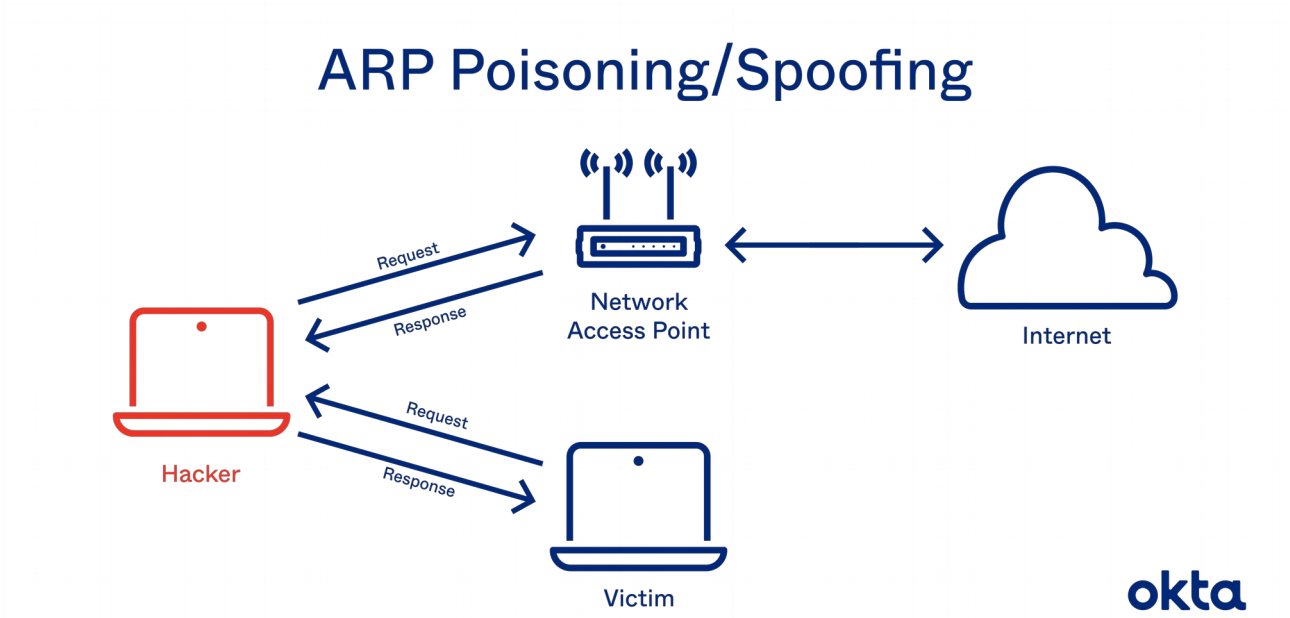
(a) Falsificação ARP

1. Definição e motivação

Falsificação ARP (ARP spoofing) é um tipo de ataque onde o autor envia mensagens ARP falsas para uma rede local, causando a conexão entre o MAC do atacante e o endereço IP de uma vítima da rede. Após a conexão, o atacante começa a receber dados que seriam destinados ao IP atacado, como se esse IP agora pertencesse ao MAC do atacante.

Esse ataque permite interceptação, modificação ou interrupção de dados. Ele também pode abrir caminho para outros ataques como DoS, sequestro de sessão e MITM (Man-in-the-middle).

Fonte: https://www.veracode.com/security/arp-spoofing



Fonte: https://www.okta.com/identity-101/arp-poisoning/

2. Funcionamento

1) O atacante abre uma ferramenta de falsificação ARP (Arpspoof, Cain & Abel, Arpoison, Ettercap, etc...) e define um IP para essa ferramenta de forma que esse IP seja o mesmo da subrede a ser atacada.
2) O atacante usa a ferramenta para escanear os IPs e MACs dos dispositivos conectados na subrede.
3) O atacante escolhe uma vítima e começa a enviar pacotes ARP pela rede, contendo o MAC do atacante e o IP da vítima.
4) Agora, todos os dispositivos da rede contém o IP da vítima associado ao MAC do atacante. Sendo assim, qualquer dado enviado para esse IP irá para o atacante ao invés da vítima.

Fonte: https://www.veracode.com/security/arp-spoofing

3. Detecção e/ou prevenção

Através da detecção passiva, o tráfego de pacotes ARP é monitorado através de programas (Arpwatch, ARP-GUARD, XArp, Wireshark). Na ativa, o administrador do sistema causa um ataque de falsificação ARP na própria rede para averiguar possíveis pontos fracos nela.

Filtragem de pacotes pode bloquear pacotes com informações conflitantes de endereço de origem (quando um pacote de fora da rede possui um endereço de algum dispositivo de dentro da rede).

Criptografar todas as comunicações ajuda a se proteger desses ataques, usando protocolos de comunicação como TLS, SSH ou HTTPS. O uso de VPN também é uma fonte de proteção, pois ele cria uma conexão encriptada da origem ao destino.

Outro método de prevenção é utilizar ARP estático, definindo o cache de ARPs, que possui MAC e IP de cada dispositivo da rede. Isso vai permitir o mapeamento permanente entre MAC e IP. Ou seja, cada MAC estará preso a um IP.

Fontes:https://www.okta.com/identity-101/arp-poisoning/
https://www.veracode.com/security/arp-spoofing

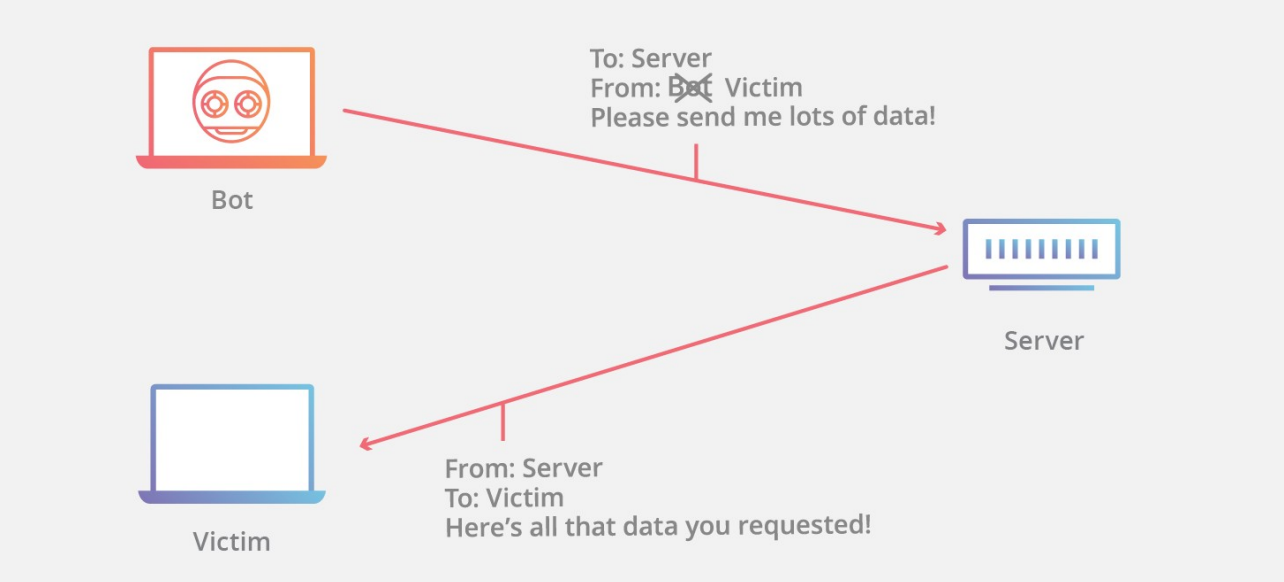
(b) Falsificação IP

1. Definição e motivação

Falsificação de IP consiste em modificar o IP de origem de um pacote para que o atacante não seja identificado, ou se passe por outro dispositivo, ou os dois.

Essa falsificação é usada para promover ataques DDoS contra um dispositivo ou à infraestrutura onde esse dispositivo está.

Fonte: https://www.cloudflare.com/pt-br/learning/ddos/glossary/ip-spoofing/

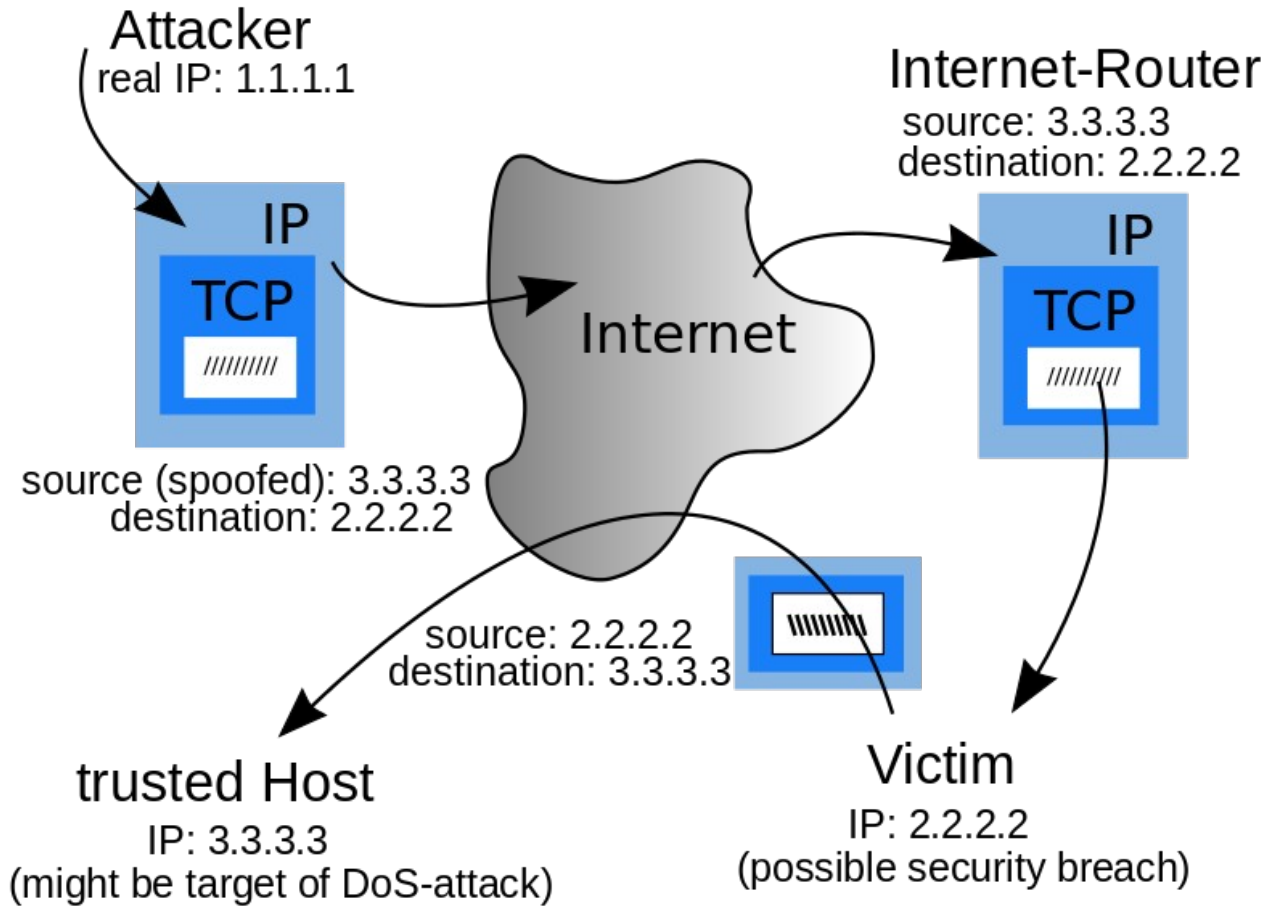


Fonte: https://www.cloudflare.com/pt-br/learning/ddos/glossary/ip-spoofing/

2. Funcionamento

   O falsificador altera o endereço IP de origem no cabeçalho do pacote IP (usado como regra de comunicação entre dispositivos na internet) e envia esse pacote na rede até chegar ao destino. O destino verá o IP de origem (falsificado) e mandará a resposta para ele.

   Fonte: https://www.blockbit.com/pt/blog/o-que-e-ip-spoofing/



Fonte: https://en.wikipedia.org/wiki/IP_address_spoofing

3. Detecção e/ou prevenção

   É possível se prevenir através de filtragem de entrada e/ou saída. Na filtragem de entrada, o verificador (pode ser algum firewall) examina o cabeçalho dos pacotes de IP que estão entrando na rede, em busca do IP de origem e compara com o IP real. Na filtragem de saída o verificador examina o cabeçalho dos pacotes para averiguar se o IP de origem é legítimo.

   Com o firewall é possível criar uma zona de proteção, onde o sistema só aceita pacotes originados de endereços IPs específicos, definido pelo administrador da rede. Em alguns sistemas o usuário só acessa a rede se estiver autenticado no firewall. Isso garante que o dispositivo seja sempre identificado.

   Fontes: https://www.blockbit.com/pt/blog/o-que-e-ip-spoofing/
   https://www.cloudflare.com/pt-br/learning/ddos/glossary/ip-spoofing/

(c) Ataque de previsão de sequência

1. Definição e motivação

   É um ataque de falsificação de IP junto com MITM (Man-in-the-middle). Durante a conexão entre cliente e servidor (TCP handshake), o atacante “se posiciona” entre esses dois, passando a monitorar toda a comunicação. Falsificando o IP, ele consegue ser identificado como cliente ou como servidor, passando a fazer parte da conexão. A partir de então, o atacante consegue encerrar a conexão entre cliente e servidor, ou acessar dados importantes, ou executar script maliciosos em ambos os lados.

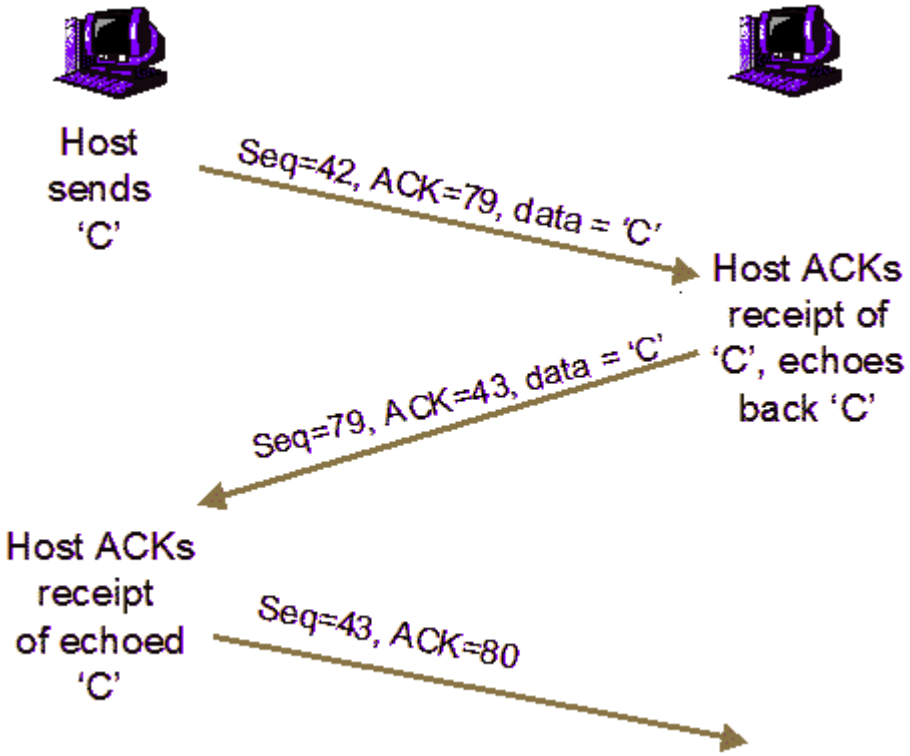
   Fonte: https://www.information-age.com/end-threat-sequence-prediction-attacks-123463896/

2. Funcionamento

   O protocolo TCP garante que a conexão entre origem e destino seja feita. Isso é feito através da numeração de pacotes. Cada pacote tendo um número subsequente ao anterior. O MITM explora essa ordem de pacotes executando os seguintes passos:

   1) O atacante escuta a comunicação e define um alvo.
   2) O atacante emite pacotes ao alvo usando o endereço IP de origem de quem está se comunicando com o alvo. Ex: Entre cliente e servidor, o atacante envia pacotes ao cliente usando o endereço IP do servidor.
   3) Esses pacotes devem ter o número que o alvo está esperando e devem chegar antes dos pacotes enviados pelo agente confiável. Geralmente é usado ataque DoS para atrasar o agente. Desta forma, os pacotes do atacante chegam antes dos pacotes do host confiável.
   4) Agora a conexão entre atacante e alvo foi estabelecida. É possível se comunicar usando uma conexão TCP/IP comum.

   Fonte: https://www.idc-online.com/technical_references/pdfs/data_communications/TCP_Sequence_Prediction_Attack.pdf



Fonte: https://www.tech-faq.com/tcp-sequence-prediction-attack.html

3. Detecção e/ou prevenção

   Esses ataques podem ser parados por um roteador ou firewall configurados para não permitirem pacotes com endereços IP internos vindos do ambiente externo.
   Também é possível se proteger usando criptografia entre a origem e o destino da conexão (usando protocolo SSL, que encripta toda a comunicação entre websites e o browser do usuário).
   Fontes:https://www.information-age.com/end-threat-sequence-prediction-attacks-123463896/
   https://www.idc-online.com/technical_references/pdfs/data_communications/TCP_Sequence_Prediction_Attack.pdf

(d) Sequestro de sessão TCP

1. Definição e motivação

   Feito após um ataque de predição de sequência, o atacante assume a comunicação entre um alvo. Comunicação esta, que era feita entre alvo e algum outro host. Após o sequestro da sessão, o atacante pode fingir ser esse host confiável e passa a receber qualquer informação que seria destinada ao usuário confiável.

   Fonte: https://pages.uoregon.edu/stevev/cis399/notes/network-attacks/09.html

2. Funcionamento

   Um ataque de predição de sequência é feito e então a conexão entre alvo e atacante é estabelecida.
   Fonte: https://pages.uoregon.edu/stevev/cis399/notes/network-attacks/09.html

3. Detecção e/ou prevenção

   A evolução do protocolo TCP tem trazido incrementos na implementação, onde a sequência inicial de números aleatórios que compõem a sequência de pacotes para ordenação deles após a comunicação, passou a ser definido usando um gerador de

números pseudoaleatórios criptográficos. Desta forma, o atacante não consegue prever o próximo número.

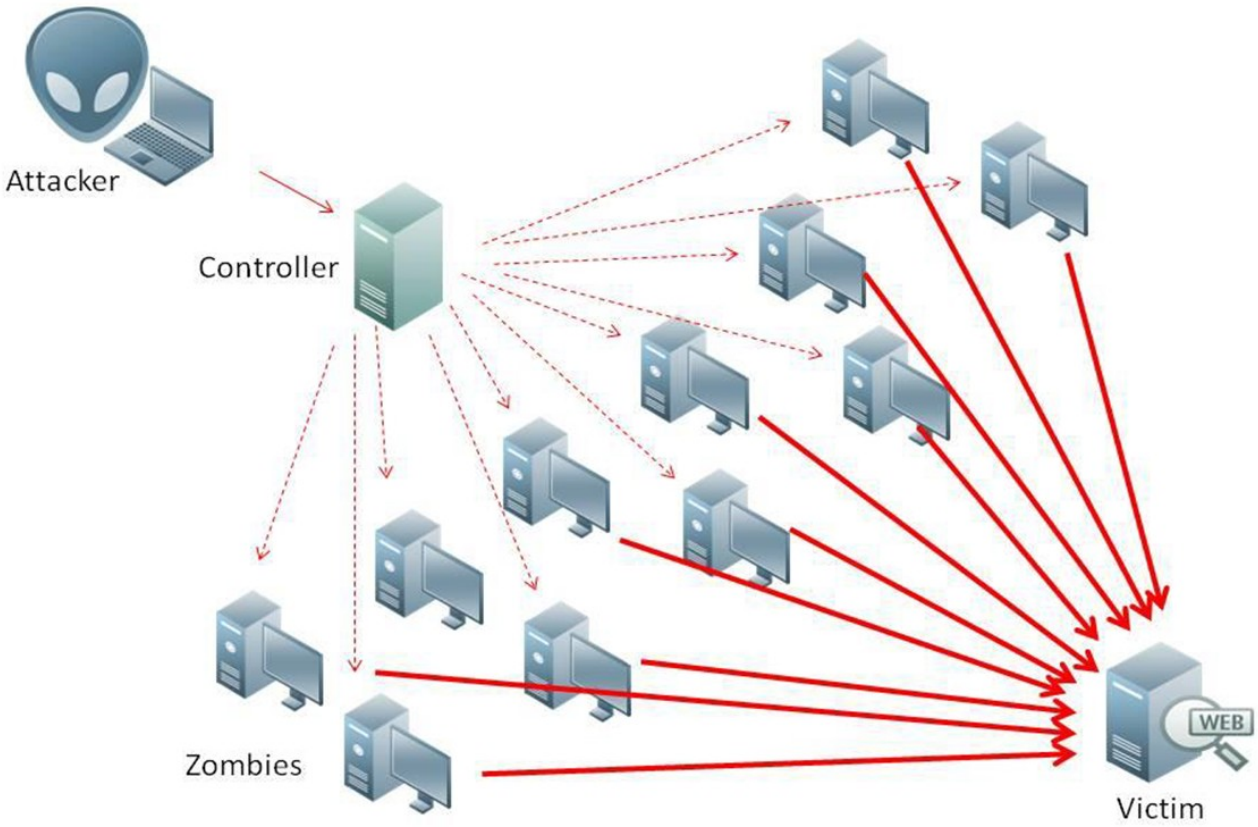
Fonte: https://pages.uoregon.edu/stevev/cis399/notes/network-attacks/09.html

(e) Ataque Smurf

1. Definição e motivação

É um ataque DDoS com o objetivo de derrubar uma rede de computadores.

Fonte: https://www.kaspersky.com.br/resource-center/definitions/what-is-a-smurf-attack

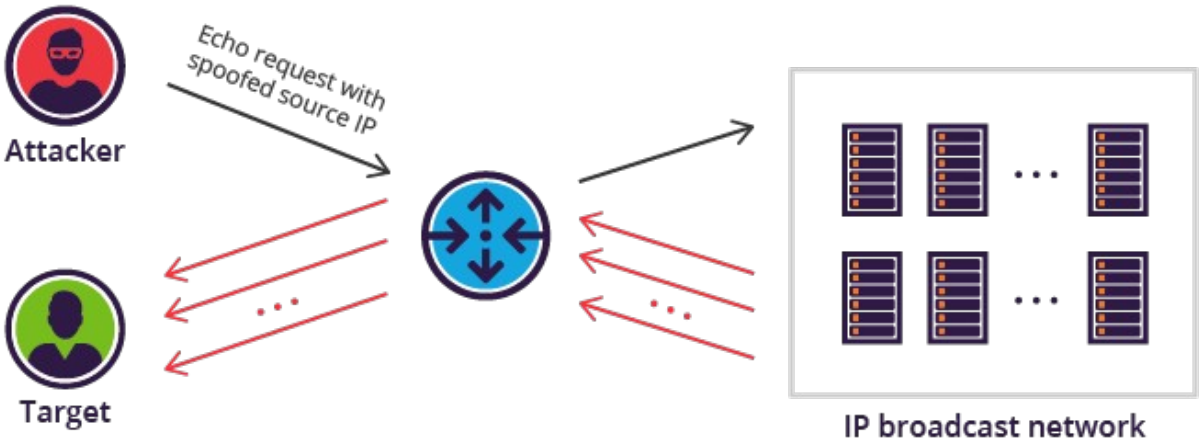


Fonte: https://pt.wikipedia.org/wiki/Ataque_Smurf

2. Funcionamento

1) O atacante faz uma falsificação de IP.
2) O pacote com o IP falsificado possui um comando de ping por ICMP, solicitando que os dispositivos que receberam esse pacote, enviem uma reposta ao IP falsificado.
3) As respostas são feitas indefinitivamente até derrubar a rede.

Fonte: https://www.kaspersky.com.br/resource-center/definitions/what-is-a-smurf-attack



Fonte: https://pt.wikipedia.org/wiki/Ataque_Smurf

3. Detecção e/ou prevenção

Bloqueio do tráfego de transmissão direcionada que chega à rede.

Configuração de hosts e roteadores para que não respondam a solicitações de eco do ICMP.

Fonte: https://www.kaspersky.com.br/resource-center/definitions/what-is-a-smurf-attack

Q2

O login remoto funciona da mesma forma que um compartilhamento de desktop. O computador hospedeiro permite que um usuário remoto acesse conteúdo e controle de mouse e teclado do computador hospedeiro através da internet. É necessário que ambos os computadores possuam o mesmo software de compartilhamento.

Fonte: https://computer.howstuffworks.com/how-desktop-sharing-works1.htm

As diferenças entre Telnet (Telecommunications and Networks) e SSH (Secure Shel) são as seguintes:

- Telnet é um protocolo de rede desenvolvido especificamente para redes locais. SSH é um programa usado para fazer conexão em um computador remoto, executar comandos e mover arquivos de uma máquina a outra.
- Telnet usa a porta 23 e o SSH usa a porta 22 (padrão) e pode ser facilmente modificada.
- Telnet é menos seguro que o SSH pois este compartilha informações criptografadas entre as máquinas.
- Telnet transfere dados em formato .txt. SSH usa dados encriptados e, também, um canal de segurança.
- Telnet não provê privilégios nem autenticação. SSH usa criptografia com chave pública para autenticação.
- Telnet não é recomendado para redes públicas, ao contrário do SSH.

Fonte: https://www.tutorialspoint.com/difference-between-ssh-and-telnet

SSH pode usar criptografa simétrica ou assimétrica ou hashing.

Na simétrica, uma chave é compartilhada durante a codificação e decodificação de mensagens entre cliente e servidor. Cada sessão SSH possui sua chave. Para isso, é necessário autenticação prévia do cliente para usar a máquina externa ou servidor.

Para assimétrica, é usada duas chaves (pública e privada). Uma para codificar e outra para decodificar. A chave pública é compartilhada entre cliente e servidor, mas a privada, não. A mensagem é codificada pela chave pública e decodificada pela chave privada. Ou seja, não é possível codificar e decodificar usando a mesma chave.

O hashing trabalha como uma criptografa de via única. Ele encripta, não sendo possível a decriptação da mensagem. O hashing gera um valor único de comprimento fixo para cada entrada. Se a entrada possui uma letra trocada, a saída gerada será toda diferente. O SSH usa essas saídas para autenticar a validade das mensagens.

Fonte: https://www.weblink.com.br/blog/tecnologia/acesso-ssh-o-que-e/

Q3

IPSec funciona da seguinte forma:

1. É feita a troca de chaves para que seja possível a codificação e decodificação entre as duas partes.
2. Quebra dos dados em vários pacotes, cada um com cabeçalhos e trailers contendo instruções sobre o que tem no pacote, de onde vem, para onde vai, selo de autenticidade e informação sobre encriptação.
3. Cada pacote possui um selo de autenticidade para comprovar a confiabilidade da origem.
4. É feita a encriptação do dado útil para cada pacote e cada cabeçalho.
5. É feita a transmissão dos pacotes através da rede. IPSec utiliza UDP como protocolo de transporte.
6. Por fim, é feita a decriptação dos pacotes.

Fonte: https://www.cloudflare.com/pt-br/learning/network-layer/what-is-ipsec/

VPN funciona da seguinte forma: Após se conectar ao servidor VPN, é criado um “túnel” encriptado por onde os dados do usuário irão fluir. A partir de então todas os acessos (comunicações) pela rede feita pelo usuário serão registradas como se fossem feitas pelo servidor da VPN e isso inclui o endereço IP.

Fonte: https://www.namecheap.com/vpn/how-does-vpn-virtual-private-network-work/

Q4

Sim. Levando em consideração a velocidade atual dos processadores, é possível executar o cálculo de contagem de IPs em segundos ou talvez até menos de 1 segundo, dependendo da quantidade de fluxo da rede e do processador no servidor. Com os processadores atuais, com velocidade por volta de 4GHz e 4 ou 8 núcleos seria muito viável. Levando em consideração que o IPv4 possui 2^{32}(cerca de 4 bilhões [exatamente 4.294.967.296]) endereços possíveis, um processador de 4GHz de 2 núcleos faz algo em torno de 8 bilhões de cálculos por segundo (a grosso modo, porque depende da tecnologia embarcada nesse processador), o que seria quase o dobro de endereços IPv4. E ainda deve-se levar em consideração que dificilmente um servidor vai receber todos os endereços IPv4 do mundo para processar, pelo menos não por enquanto.