

Introdução

Segurança de Redes: deter, prevenir, detectar e corrigir violações de segurança que envolvem a transmissão de informação

Segurança de Computadores: proteção colocada num sistema de informação automatizado de maneira a atingir objetivos aplicáveis de preservar a CIA dos recursos do sistema de informação

Confidencialidade: garante que indivíduos controlam ou influenciam que informação relacionada com eles pode ser coletada e armazenada, por quem e para quem a informação pode ser disponibilizada

Integridade: garante que um sistema realiza a sua função pretendida de forma intacta, livre de manipulação desautorizada (deliberada ou inadvertida) do sistema

Disponibilidade: garante que os sistemas funcionam de acordo com as suas especificações operacionais e que o serviço não é negado a utilizadores autorizados

Autenticidade: verificação de que os utilizadores não querem eles dizerem ser

Ameaga: conta quem se querem proteger os dados — quem pode aceder?

Encriptação: MENSAGEM + CHAVE → TEXTO CIFRADO

Desencriptação: TEXTO CIFRADO + CHAVE → MENSAGEM

Controlo de Acesso: regras e políticas que limitam o acesso a informação confidencial a pessoas e/ou sistemas numa base de "need to know"

Ferramentas de Integridade:

1. Redundância: backups periódicos, idealmente armazenados em máquinas heterogéneas
2. Checksums: computar uma função que mapeia os conteúdos de um ficheiro num valor numérico
3. Códigos de Correção de Dados: têm informação adicional para corrigir erros
4. Códigos de Autenticação de Mensagens: cálculo baseado numa chave secreta
5. Assinaturas Digitais

Ferramentas de Disponibilidade:

1. Proteções Físicas: infraestrutura pode manter informação disponível mesmo no evento de desafios físicos
2. Redundância Computacional: múltiplos servidores e backends podem garantir que o serviço se mantém disponível mesmo no evento de (algumas) falhas

Autenticação: para determinar a identidade ou papel que alguém tem num sistema — algo que eu sei/tendo/sou

Ferramentas de Autenticidade:

1. Assinaturas Digitais: computações criptográficas que permitem que uma pessoa ou sistema confirmem/atestem a autenticidade dos seus documentos
- Normalmente (mas não sempre) também garantem não-refúcio — afirmações autênticas não podem ser negadas

AMEAÇAS & ATAQUES

Eavesdropping: interceptação de informação durante a sua transmissão por um canal de comunicação — confidencialidade

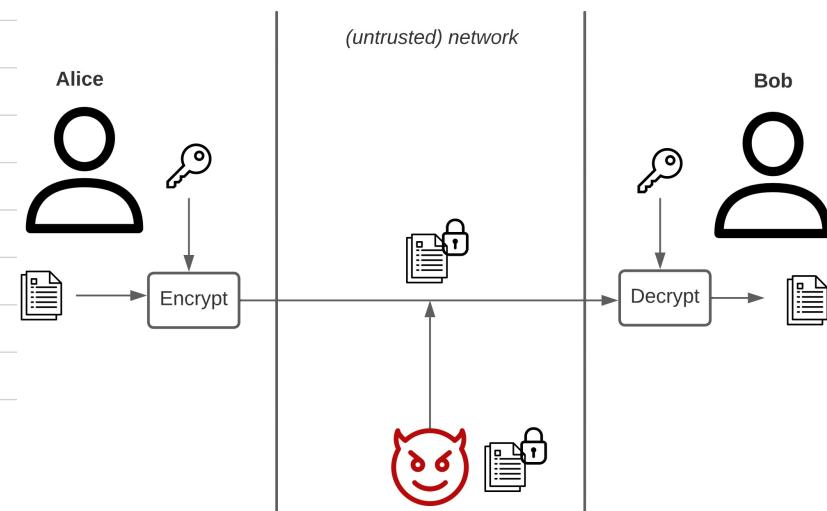
Man-in-the-Middle: interceptar um fluxo de dados, (às vezes) modificá-lo e retransmiti-lo — confidencialidade e integridade

Denial-of-Service: interromper ou degradar um serviço ao sobrepor-lhe com mensagens — disponibilidade

Dirfusor-se/Mascarar-se: fabrico de informação que é suposta ser de alguém que não é de facto o autor — autenticidade

Superfície de Ataque: vulnerabilidades alcançáveis e exploráveis num sistema

- Rede: rede empresarial ou internet
- Software: aplicação, utilitário ou código do sistema operativo
- Humana: pessoal interno ou externo



Criptografia

Criptografia: arte e ciência de estabelecer comunicação segura num canal inseguro

Confidencialidade: proteger dados sensíveis de "eavesdropping"

Integridade: detectar se as mensagens não são alteradas

Criptografia Simétrica
Códigos de Autenticação de Mensagens

VS. Criptografia de Chave Pública
VS. Assinaturas Digitais

Criptografia Simétrica: os utilizadores usam a mesma chave (pré-partilhada)

Encriptação Simétrica: C

Encriptação: $c = E(k, m)$

Desencriptação: $m = D(k, c)$

\rightarrow AES-CBC; AES-CTR

Encriptação Autenticada com Dados Associados (AEAD): AES-GCM; Poly-ChaCha

Códigos de Autenticação de Mensagens: I

Autenticação: $t = \text{MAC}(k, m)$

Verificação: T/F = $V(k, m, t)$

\rightarrow HMAC; CMAC

Criptografia de Chave Pública: os utilizadores trabalham com chaves diferentes

Encriptação: C & I

Encriptação: $c = E(pk, m)$

Desencriptação: $m = D(sk, c)$

\rightarrow RSA-OAEP

Assinaturas Digitais: I & NR

Autenticação: $t = S(sk, m)$

Verificação: T/F = $V(pk, m, t)$

\rightarrow Schnorr

Protocolos de Troca de Chave: troca de chave simétrica; Diffie-Hellman

Infraestrutura de Chave Pública: A e B podem não confiar em CA_A nem CA_B, mas confiam em CA_{ROOT}, que certifica outras CAs, que certificam chaves públicas - hierarquia de confiança

\rightarrow Base de Computação Confável

Autenticação: determinar se um utilizador deve ter acesso a um sistema **PROTÓCOLOS**
Autorização: decidir que ações um utilizador pode realizar num sistema

Protocolo de Rede: regras seguidas em sistemas de comunicação em rede ***TP***
Protocolo de Segurança: regras (de comunicação) seguidas numa aplicação de segurança ***S***

O que torna um protocolo confiável?

1. Satisfaz requisitos de segurança
2. Eficiente - minimiza recursos computacionais, uso de largura de banda e armazenamento
3. Robusto - funciona quando um atacante tenta quebrá-lo e em ambientes instáveis
4. Fácil de implementar e operar
5. Flexível em configuração e produção

⚠ → O adversário pode observar mensagens previamente e reproduzi-las

Ataque de Repetição: o adversário observa a interacção e usa as mensagens para repetir um padrão de comunicação

SOLUÇÃO → Desafio-Resposta: se B quer autenticar A, então B envia um desafio para A e A deve responder ao desafio de acordo com a sua palavra-passe

Desafio: escolhido de modo que a repetição não seja possível, só A pôde fornecer a resposta correta e B pôde verificar a resposta (eficientemente)

Nonce: número que só é usado uma vez

Desafio: nonce - cada pedido para autenticação deve usar um nonce diferente

Resposta: hash - a mensagem usada para a primeira autenticação não vai funcionar para nenhuma das seguintes

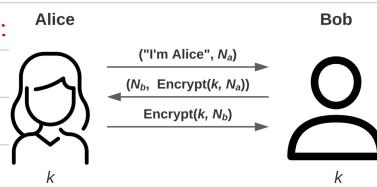
$$h(\text{password} \parallel \text{nonce})$$

$$c \leftarrow E(k, \text{nonce})$$

Chave Simétrica: A e B partilham a chave simétrica k e não os únicos que a conhecem — autenticação requer provar o conhecimento de k

Autenticação Mútua:

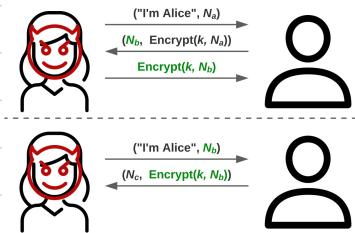
1. B autentica A
2. A autentica B



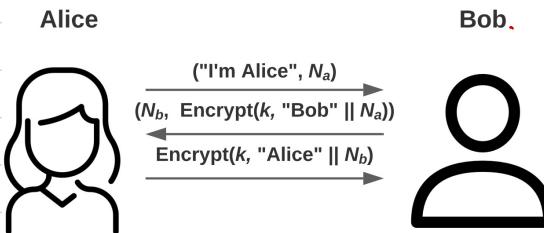
ATAQUE
REFLEXÃO

Alice (?)

Bob

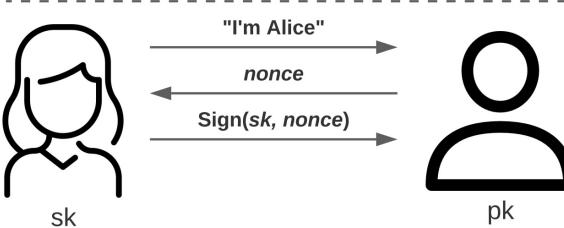
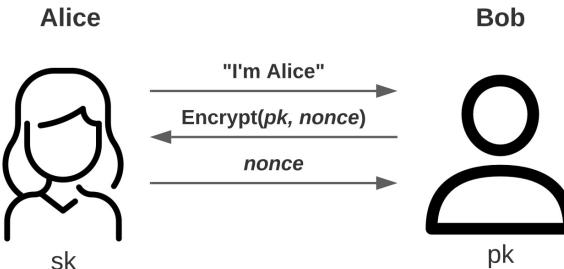


Ataque de Reflexão: a resposta não está limitada a uma execução específica do protocolo — Como garantir relação 1-1 entre respostas, desafios, identidades?



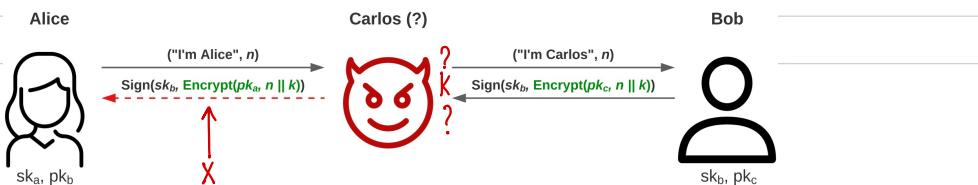
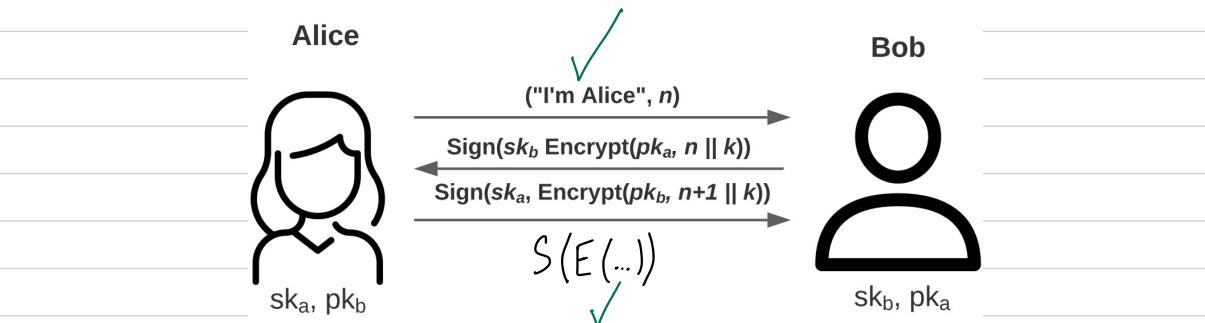
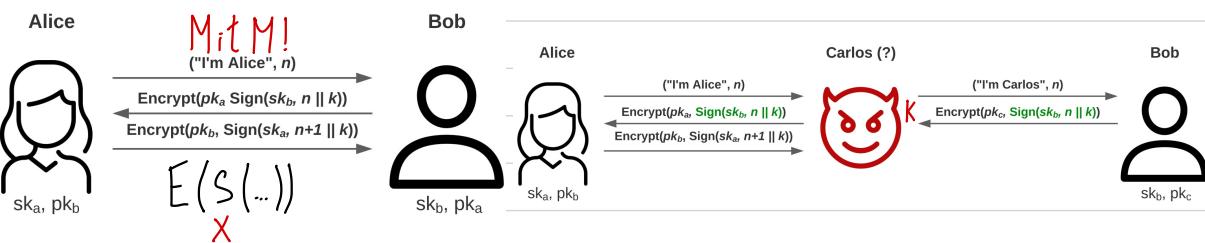
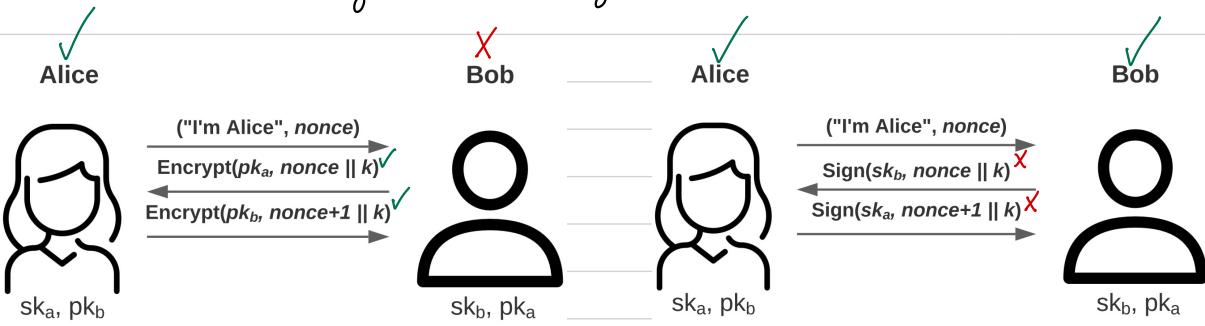
Chaves Assimétricas: A tem chave secreta sk e B tem chave pública pk e só A conhece sk — autenticação requer provar o conhecimento de sk

Boa-Prática: usar um par de chaves para E/D ou S/V e outro para autenticação



Autenticação

Chave de Senão: chave simétrica para cada senão — efêmera
 ↳ confidencialidade/integridade



Kerberos

Kerberos: protocolo de autenticação que confia numa Terceira Parte Confável (TPC)

Autenticação com Chave Pública: N utilizadores $\rightarrow N$ pares de chaves

Autenticação com Chave Simétrica: N utilizadores $\rightarrow \approx N^2$ chaves — não escala

Autenticação com Kerberos: N utilizadores $\rightarrow N$ chaves (simétricas)

Centro de Distribuição de Chaves (KDC): TTP que não pode ser comprometida (por armadilha) e que partilha chaves simétricas com todos os participantes e uma chave-mestre extra (KDC), permitindo autenticação e chaves de sessão

→ Concede Tickets necessários para aceder aos recursos da rede

→ Concede Tickets que Concedem Tickets (TGT) para obter tickets regulares

Tickets que Concedem Tickets (TGT): fornecidos no primeiro login, contêm chave de sessão, ID do utilizador e tempo de expiração, usados para garantir que o KDC gira uma base de dados, permitindo que se mantenha "stateless" (quase)

- A chave K_N é derivada/generada a partir da palavra-passe de N
- O KDC gera a chave de sessão S_N e o TGT e entrega-os
- O tempo é verificado para garantir frescura

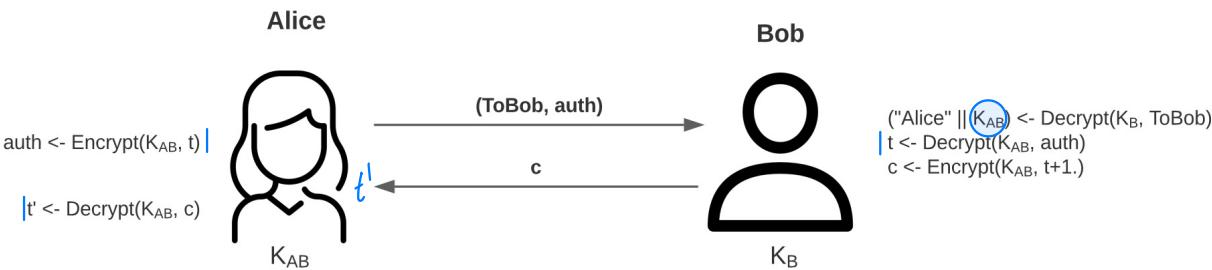
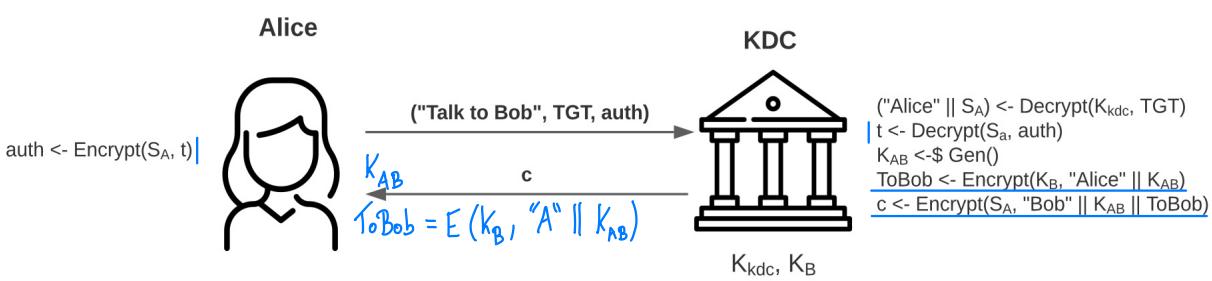
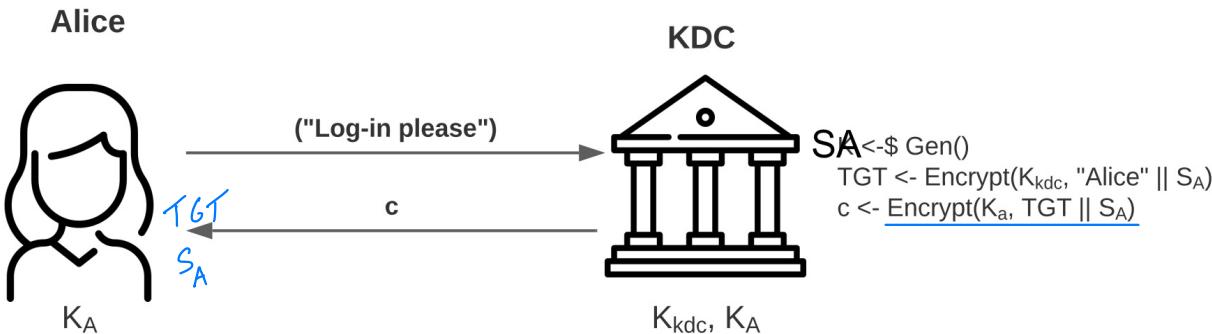
S_A : usada para autenticação — confidencialidade/integridade para $A \leftrightarrow KDC$

K_{AB} : usada para $A \leftrightarrow B$ — confiável porque o KDC a encripta com K_B
 t : "nonce" usado para autenticação e proteção contra repetição — frescura

• TGT é encriptado com K_A para anonimidade de A

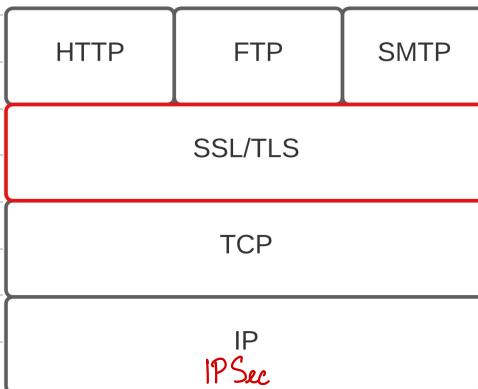
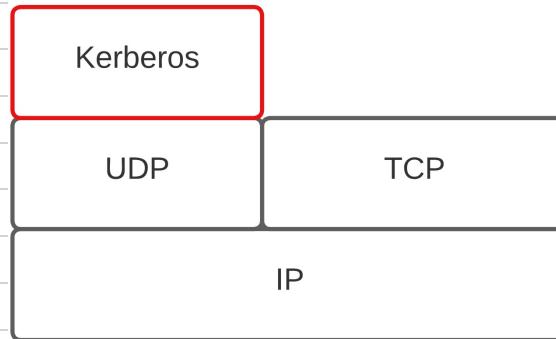
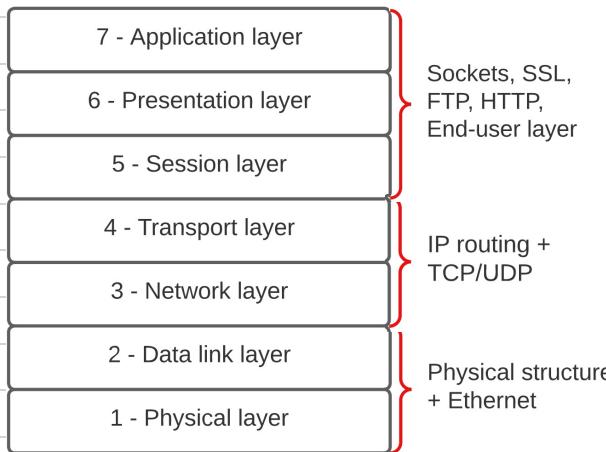
• KDC não envia o ticket para B para que ele não tenha de decorar K_{AB}

• KDC não decora a chave de sessão (em vez de usar TGT) para ser "stateless"



Segurança Web

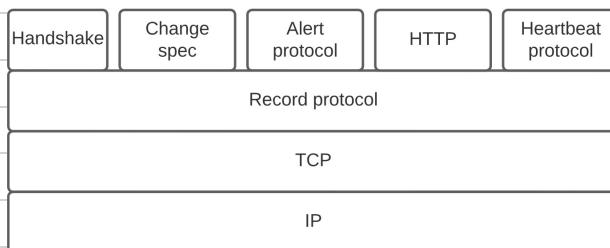
A World Wide Web é fundamentalmente uma aplicação cliente-servidor a correr sobre a internet e sobre intranets TCP/IP



SSL/TLS

SSL: Secure Sockets Layer \rightsquigarrow TLS: Transport Layer Security

SSL/TLS: sistema de propósito geral implementado como um conjunto de protocolos que assentam/confiam em TCP para assegurar garantias de entrega de mensagens



Protocolo "Record": integridade e confidencialidade - usa a chave acordada anteriormente

"Handshake": estabelece uma chave criptográfica

"Change Cipher Spec": mensagem única que estabelece as especificações de cifra acordadas

Protocolo "Alert": pode provocar avisos ou terminar conexões

Protocolo "Heartbeat": "pinga" regularmente, prevenindo a conexão de ir abaixo

Conexão TLS: transporte que fornece um tipo de serviço adequado - "peer-to-peer",
ESTADO transient e associada com uma versão

1. Aleatoriedade do Cliente e Servidor: sequências de bytes escolhidas pelo cliente e servidor
2. Chave MAC do Servidor: chave criptográfica usada para autenticar mensagens do servidor
3. Chave MAC do Cliente: chave criptográfica usada para autenticar mensagens do cliente
4. Chave do Servidor: chave criptográfica usada para encriptar mensagens do servidor
5. Chave do Cliente: chave criptográfica usada para encriptar mensagens do cliente
6. Vetores de Inicialização: para garantir frescura dos textos cifrados
7. Números de Sequência: menores do que 2^{64}

Sessão TLS: associação entre um cliente e um serviço, criada pelo protocolo "handshake", que define um conjunto de parâmetros de segurança criptográfica partilhados entre múltiplas conexões, usada para evitar etapas dispensáveis de negociação no início de cada sessão

↓ ESTADO

1. Identificador: sequência arbitrária de bytes escolhida pelo serviço para identificar
2. Certificado: certificado X509.v3 do par
3. Método de Compressão: algoritmo usado para comprimir dados antes da encriptação
4. Especificação da Cifra: algoritmos de encriptação e de hash e atributos criptográficos
5. Segredo Mestre: chave simétrica partilhada entre cliente e serviço
6. É Retomável?: "flag" que indica se a sessão pode ser usada para iniciar novas conexões

→ Protocolo "Record"

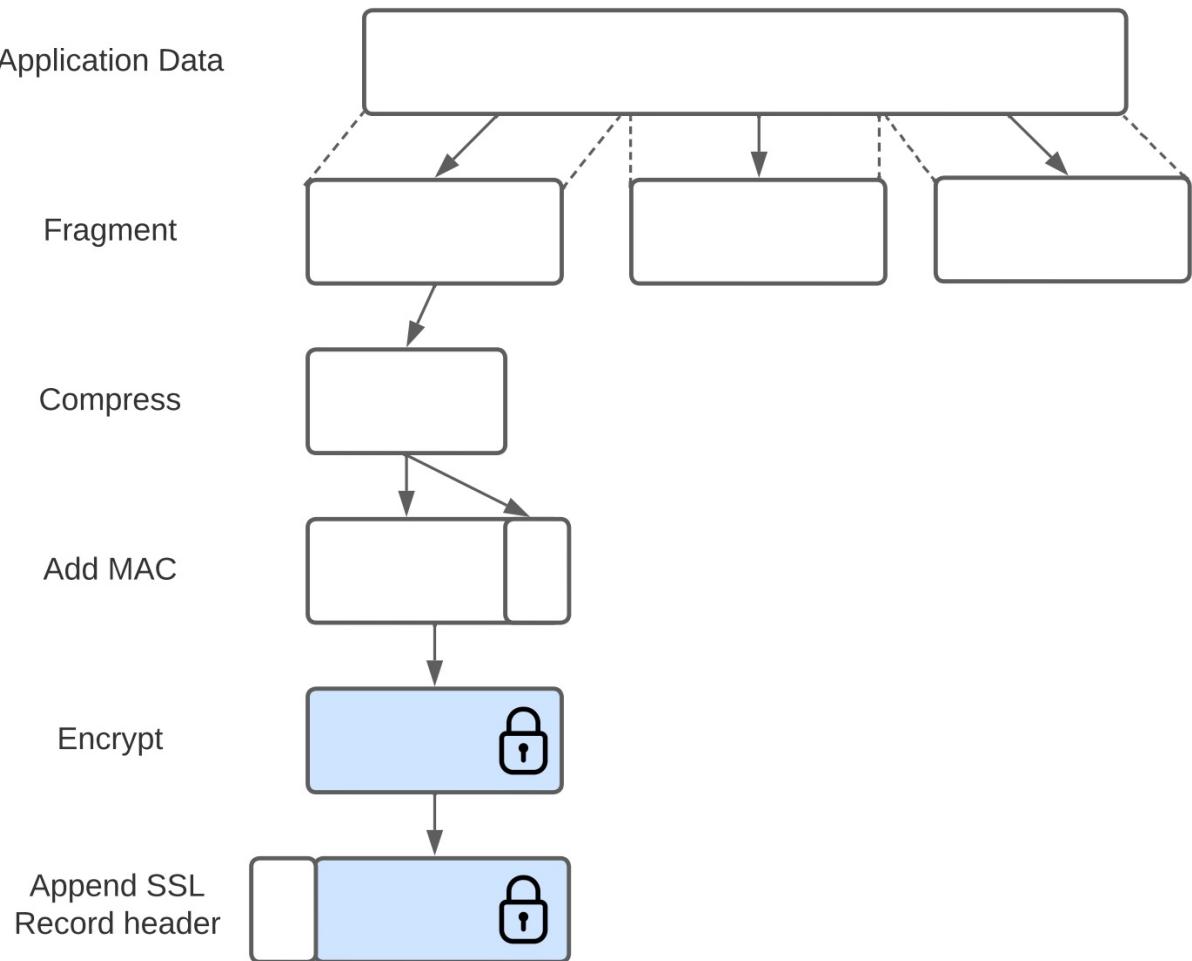
- | | | | | |
|-----------|----------------|-------------|-----------------|-------------|
| Emissor: | 1. Fragmenta | 2. Comprime | 3. Adiciona MAC | 4. Encripta |
| Receptor: | 1. Desencripta | 2. Verifica | 3. Descomprime | 4. Monta |

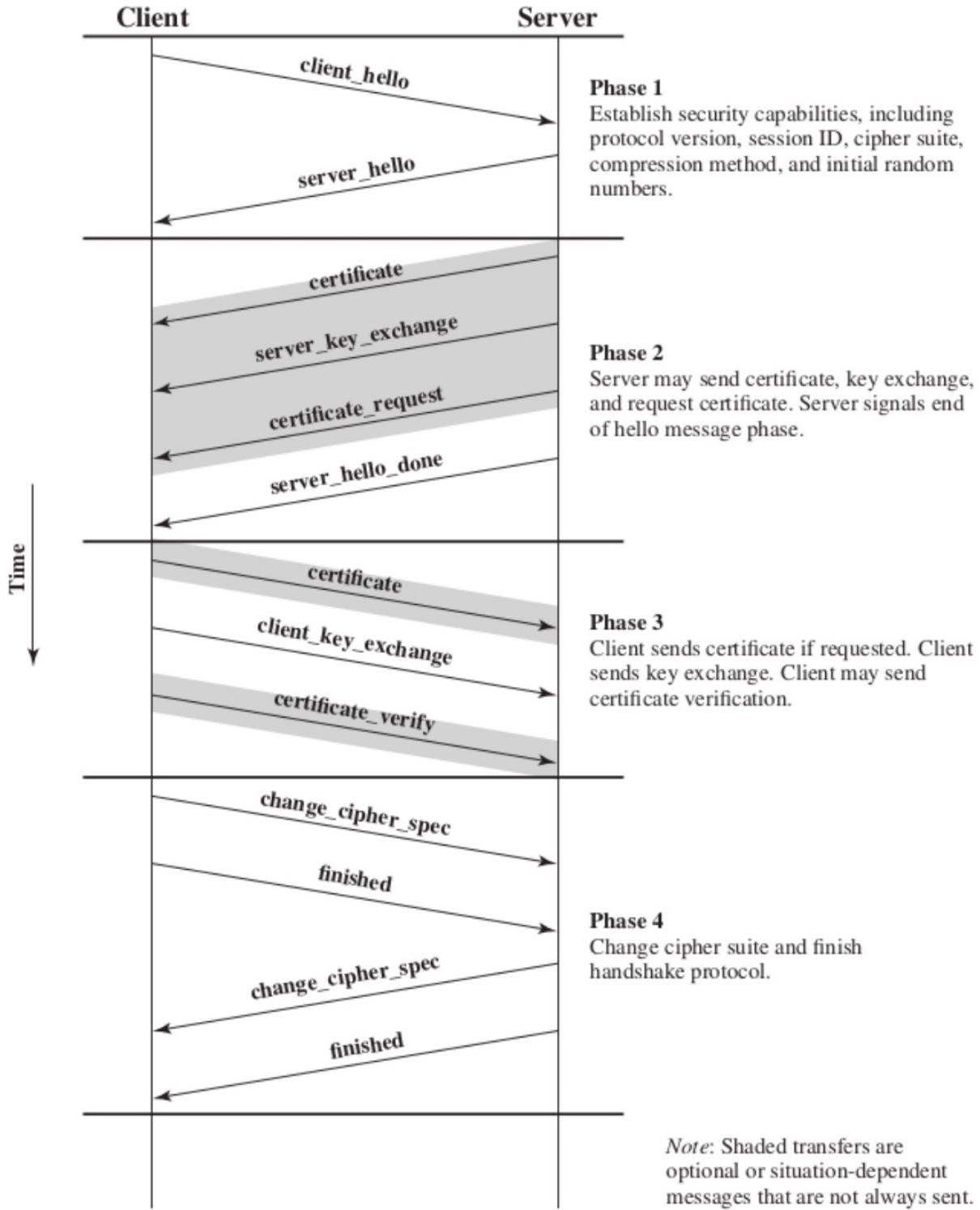
→ Protocolo "Handshake": usado antes de quaisquer dados serem transmitidos, permite autenticação mutua e negociação dos algoritmos de encriptação, de MAC e chaves

1. Hello! — versão TLS, ID da sessão, Cipher Suite, método de compressão
- 2./3. Troca e verificação de certificados e acordo da chave
4. Cliente envia especificações de cifra e mensagem final protegida com encriptação autenticada, usando os novos algoritmos, chaves e negociações — Serviço verifica e faz o mesmo

→ Protocolo "Change Cipher Spec": mensagem única de um único byte cujo valor é 0 ou 1 — confirmação que toma o estado pendente atual e atualiza a cifra em uso

→ Protocolo "Alert": mensagem comprimida e encriptada de dois bytes em que o primeiro se refere à gravidade e o segundo especifica — mensagens fatais terminam a conexão imediatamente **OU** outras conexões para a sessão podem continuar, mas não podem ser estabelecidas conexões adicionais





→ Protocolo "Heartbeat": sinal periódico gerado por hardware ou software para indicar operação normal ou sincronizar com outras partes do sistema - monitorização

HEARTBEAT REQUEST: "prova que estás vivo" - comprimento; payload; padding
HEARTBEAT RESPONSE: "eu estou, de facto, vivo" - payload

Objetivo: assegura o emissor de que o receptor ainda está vivo e impede fecho automático da conexão pelo firewall para conexões sem resposta

1. Envia mensagem
2. Escreve, prepara "payload" e envia resposta
3. A resposta contém exatamente o tamanho esperado do "payload"
4. Verifica validade do "payload"

Heartbleed:

1. Envia "payload" pequeno disfarçado de grande
2. Escreve, prepara "payload" (mau) e envia resposta
3. A resposta contém muito mais do que o esperado
4. Obtém chaves TLS, cookies e palavras-passe!

HTTPS

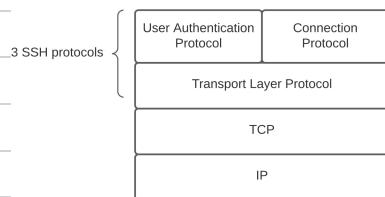
HTTPS: HTTP sobre SSL - combinação de HTTP e SSL para implementar comunicação segura entre navegador Web e servidor Web

Protege: 1. URL 2. Conteúdos do documento, formulários e cabeçalho 3. Cookies

- Todos os dados são enviados como dados aplicacionais TLS
- A conexão é fechada com a linha CONNECTION: CLOSE
- A implementação TLS troca alertas de fecho antes de fechar a conexão

SSH

SSH: Secure Shell Protocol — fornece um caminho encriptado e autenticado à linha de comandos do SO através da rede, protegendo contra ataques de "spoofing" e de modificação dos dados — método para aceder a recursos remotos



Protocolo da Camada de Transporte: fornece autenticação do serviço, confidencialidade e integridade

1. Acordo das versões do software e protocolo
2. Troca dos algoritmos suportados
3. Fim da troca de chaves
4. Serviço pronto a executar

Protocolo de Autenticação do Utilizador: autentica o utilizador do lado do cliente para o servidor

Chave Pública: cliente envia mensagem assinada com a chave privada para o servidor

Palavra-Passe: cliente envia mensagem com a palavra-passe, encriptada pelo PCT

"Host-based": cliente envia uma assinatura criada com a chave privada do anfitrião

Protocolo de Conexão: multiplica o túnel encriptado em vários canais lógicos

Sessão: execução remota de um programa

X11: GUI para computadores em rede

ICPIP encaminhado: encaminhamento remoto de portas (do PC remoto para o PC local)

ICPIP direto: encaminhamento local de portas (conversão TCP insegura → túnel SSH)

IPSec

IPSec: IP Security - segurança na camada de rede, transparente às aplicações

Autenticação: garante que um pacote recebido foi transmitido como a origem no cabeçalho do pacote e que o pacote não foi alterado em trânsito

Confidencialidade: permite que os nós de comunicação encriptem mensagens para prevenir "eavesdropping" por terceiras partes

Gestão de Chaves: garante que os nós comunicantes podem trocar de forma segura material criptográfico (chaves)

O IPSec triunfa em aplicações em que a mesma segurança é sempre necessária e as mesmas técnicas de segurança podem ser aplicadas a todas as aplicações

quando implementado numa firewall ou router, fornece segurança a todo o tráfego que crega o perimetro
transparente às aplicações e utilizadores finais

segura arquitetura de rotas

Encapsulated Security Payload (ESP): função combinada para autenticação/encriptação

Authentication Header (AH): função só de autenticação

Serviços: controlo de acesso; integridade "connectionless"; autenticação da origem dos dados; rejeição de pacotes replicados (integridade da sequência parcial); confidencialidade

Arquitetura:

1. Gestão de Troca de Chaves - protocolo IKE
2. Extensões de Segurança do Cabeçalho - AH e ESP
3. Modos de Operação - Transporte e túnel

Modo Transporte: adiciona informação/segurança ao pacote original

Modo Túnel: protege o pacote original ao encapsulá-lo num novo pacote IP

IKE

IKE: Internet Key Exchange

Phase 1: associação de segurança IKE ≈ versão - "handshake", parâmetros, negociação
Phase 2: associação de segurança IPsec ≈ conexão - efêmera, seleção de chaves

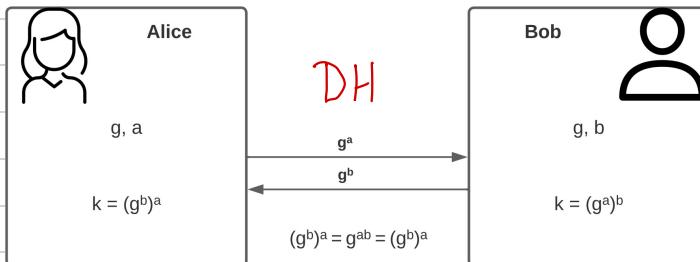
FASE 1

Opcionais:

1. Encriptação de Chave Pública (versão original)
2. Encriptação de Chave Pública (versão melhorada)
3. Assinatura de Chave Pública
4. Chave Simétrica

Modo Principal: "handshake" com encriptação autenticada

Modo Agressivo: ambos os pares têm endereços IP externos dinâmicos



1. Cookies impõem ataques de entupimento
2. Especifica os parâmetros globais usados por DH
3. Usa nonce para prevenir contra ataques de repetição
4. Permite DH trocar valores de chave pública
5. Autentica DH contra ataque de Man-in-the-Middle

Ataque de Entrapamento:

1. Um adversário forja o endereço de origem de um utilizador legítimo e envia uma chave pública DH para a vítima
2. A vítima realiza uma expoenciada modular e computa a chave
3. Mensagens repetidas entopem o sistema com trabalho inicial

Troca de Cookies:

1. Cada lado envia um número pseudo-aleatório (cookie) na mensagem original e o outro lado reconhece (ACK)
2. O ACK deve ser repetido na primeira mensagem da troca de chave DH
3. Se o endereço de origem foi forjado, então o adversário não recebe resposta

Ataque de Repetição:

os dados válidos transmitidos pelo utilizador legítimo ao receptor são capturados pelo adversário e reenviados para o autenticar

Nonces:

mechanismo de desafio-resposta

Output: autenticação mutua; chave simétrica partilhada; IKE SA

FASE 2

IPSec SA: conexão de um nó com significância local
PARÂMETROS vitais de segurança para o tráfego nela levado

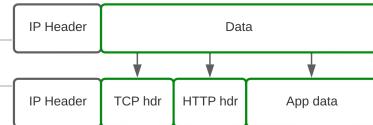
1. Índice dos Parâmetros de Segurança: número só com significância local
2. Identificador do Protocolo de Segurança: associação de segurança AH ou ESP
3. Endereço de Destino IP: endereço do "endpoint" destino da associação de segurança

Base de Dados da SA: guarda parâmetros de longo prazo associados com cada SA

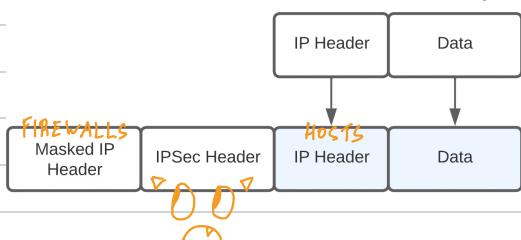
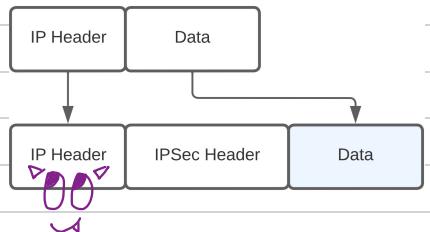
Base de Dados da SP: regras pelos quais o tráfego IP se relaciona com as SAs

R esumo

- Um datagrama IP é cabeçalho + dados
- Os roteadores têm de ver o cabeçalho, mas não têm acesso à chave da sessão
- Não se pode encriptar o cabeçalho IP!
- O tráfego web encapsula iterativamente dados



Modo Transporte: comunicação host-to-host **Modo Túnel:** comunicação firewall-to-firewall



AH: só integridade — protege tudo para além do cabeçalho e parte dele (imutável)

ESP: confidencialidade e integridade — protege tudo para além do cabeçalho, mas não protege a integridade do cabeçalho nem permite que o firewall inspecione o conteúdo

	Transport Mode SA	Tunnel Mode SA
AH	Authenticates IP payload and selected portions of IP header and IPv6 extension headers.	Authenticates entire inner IP packet (inner header + payload) plus selected portions of outer IP header and outer IPv6 extension headers.
ESP	Encrypts IP payload and any IPv6 extension headers following the ESP header.	Encrypts the entire inner IP payload.
ESP with Authentication	Encrypts IP payload and any IPv6 extension headers following the ESP header. Authenticates IP payload.	Encrypts the entire inner IP packet. Authenticates the inner IP packet.

SSH: permite diferentes canais com diferentes propriedades

IPSec: garante que (1) o anúncio de um roteador vem de um roteador autorizado, (2) um roteador procura de estabelecer/mantém uma relação de segurança com um roteador dentro domínio está autorizado, (3) uma mensagem de redireccionamento volta à sua fonte original autêntica e (4) uma atualização de rotas não é forjada

Denial-of-Service

Denial-of-Service (DoS): uma ação que previne ou impede o uso autorizado de redes, sistemas ou aplicações, ao exaurir recursos tais como CPU, memória, largura de banda e espaço em disco — forma de ataque na disponibilidade dos serviços

Largura de Banda: relaciona-se com a capacidade das ligações de rede que conectam um serviço à Internet

Recursos do Sistema: visa sobrecarregar ou quebrar o software que lida com a rede, consumindo recursos no sistema

Recursos da Aplicação: proponha vários pedidos válidos a um serviço do sistema alvo, em que cada pedido consome muitos recursos

→ "ganhá quem tem mais largura de banda"!

Ping Inundante: o objetivo do atacante é sobrecarregar a capacidade da ligação de rede à organização da vítima — o desempenho da rede é consideravelmente afetado, mas a origem do ataque é claramente identificada, a não ser que seja usado um endereço "spoofed" (servidores "zombie")

Inundação ICMP: pacotes com pedidos ICMP echo

Inundação UDP/TCP: pacotes UDP diretos para um porto ou pacotes TCP

Ataque de Reflexão: o atacante envia pacotes a um serviço conhecido do intermediário com um endereço de origem "spoofed" como a vítima real, pelo que, quando o intermediário responde, a resposta é enviada para a vítima — reflete o ataque do intermediário (refletor) para a vítima

Objetivo: gerar volume suficiente de pacotes para inundar a ligação ao sistema alvo sem alertar o intermediário

Echo-Changen: o serviço de eco (porta 07) envia de volta tudo o que recebe e Changen é um serviço de geração de caracteres
Requisito: "Spoofing" do endereço de origem

Smurf: o servidor faz "broadcast" do eco "de A" para toda a rede e A é inundada por mensagens de eco a partir de muitas máquinas

Requisito: "Spoofing" do endereço de origem

Requisito: acesso a um servidor da rede

Como estabelecer uma conexão TCP?

1. Um cliente envia uma mensagem SYN

2. O servidor responde com uma mensagem SYN-ACK

3. O cliente conclui com uma mensagem ACK

O canal só é estabelecido depois de receber ACK; até lá, espera e gasta recursos

SYN "Spoofing": o atacante envia SYN com uma origem "spoofed" (inexistente, não vai responder) e o servidor responde sucessivamente com SYN-ACK, ocupando espaço na tabela, até os pedidos de conexão falhar

Solução: cookies SYN

Metodologia de DDoS: usar múltiplos sistemas para gerar ataques

Zombie: o atacante usa uma falha no sistema operativo ou numa aplicação comum para ganhar acesso e instalar um programa

Botnet: rede de computadores infectados com software malicioso (malware) que permite que sejam controlados por atacantes

Attack-as-a-Service: serviços de Command-and-Control (C&C) são responsáveis por comandar computadores infectados, permitindo ao atacante colocar a botnet em uso

Inundação HTTP: ataque bombardeia servidores web com pedidos HTTP de muitas origens diferentes e que consomem recursos consideráveis

"Spreading": começar num link HTTP e seguir todos os links recursivamente

Slowloris: enviar pedidos HTTP legítimos que nunca completam — explora técnicas que suportam processamento paralelo de pedidos e bloqueia todas as threads não reconhecido por soluções de IDS/IPS que se baseiam em assinaturas

Amplificação DNS: usar pedidos DNS com endereço IP de origem "spoofed" atingir o alvo e explorar o comportamento do DNS (com a flag ANY) para converter um pedido pequeno (60 bytes) numa resposta muito maior (512-4000 bytes) — atacante envia pedidos a múltiplos servidores bem conectados, inundando o alvo de forma eficaz e difícil de detectar

Ataque Volumétrico: gera grandes volumes de tráfego — pressão na vítima e na infraestrutura

Alternativas:

1. Reduzir o número total de resoluçoes de DNS abertos
2. Restringir um resoluçor DNS a não responder a queries de fontes confiáveis
3. Far os ISPs a detectar ativamente endereços IP "spoofed"

"Blackhole Routing": o tráfego é encaminhado para uma rota nula e é perdido, impedindo um ataque de ser inundado — agressivo

Amplificação NTP: encontrar um servidor com bom rácio pedido-resposta e fazer "spoof" do endereço IP para encaminhar pacotes para a vítima **MONLIST**

→ Simple Service Discovery Protocol: usado por Universal Plug and Play para anunciar e procurar dispositivos/serviços na rede — UDP; M-SEARCH; SSDP:ALL; MULTICAST

Contra-medidas para Dos

1. Prevenção: antes de o ataque ocorrer - forçar políticas para o consumo de recursos e fornecer recursos de "backup" disponíveis a pedido
2. Deteção e Filtragem: durante o ataque - procurar por pacotes de componimento suspeito e filtrar pacotes do ataque
3. Rastreio e Identificação da Fonte: durante/após o ataque - identificar fontes do ataque e preparar whitelists/blacklist
4. Reação: depois do ataque - eliminar efeitos do ataque e limpar o sistema

Prevenir:

1. bloquear endereços de origem "spoofed" - o mais perto possível da origem
2. controlar trânsito em redes de distribuição - para tipos específicos de pacotes
3. lidar com conexões TCP de forma modificada - cookies SYN ou "drop" relativo/aleatório
4. bloquear pacotes IP dirigidos a "broadcast"
5. bloquear serviços suspeitos e combinações
6. usar serviços espelhados e réplicas quando for necessário desempenho e ^{fisiabilidade}
7. distinguir pedidos legítimos de humanos e de "bots"

Responder: ter um bom plano de resposta a incidentes com detalhes em como contactar o pessoal técnico do ISP e como responder ao ataque

Diagnosticar/Responder:

1. identificar o tipo de ataque - capturar e analisar pacotes e projetar filtros
2. ter o ISP a rastrear o fluxo de pacotes de volta até a origem
3. implementar um plano de contingência - mudar para servidores de "backup" alternativos
4. atualizar o plano de resposta a incidentes

Firewalls

Firewall: decide o que deixar entrar/sair na rede interna — controlo de acessos à rede

Política de Acesso da Firewall: critério que filtra por tipo de tráfego, intervalos de endereços, protocolos, aplicações e conteúdos — especificações de que tráfego tem de ser suportado

Uma má configuração pode levar a perda de comunicação

Capacidades:

1. Define um ponto único de estrangulamento
2. Fornecê uma localização para monitorizar eventos de segurança
3. Plataforma conveniente para várias funções de internet (como NAT)
4. Pode servir a plataforma para IPsec (modo túnel)

Limitações:

1. Não consegue proteger contra ataques que contornam a firewall
2. Pode não proteger totalmente contra ameaças internas
3. Dispositivos portáteis podem ser infectados fora da rede e usados internamente
4. LAN sem fios inadvertidamente protegidas podem ser acessadas fora da organização

Filtros de Pacotes: camada de rede — observa pacotes IP e avalia a sua importância

↓ LISTA DE CONTROLO DE ACESSOS

Critérios: tudo o que é visto no cabeçalho do pacote

1. endereço IP de origem/destino — permite excluir fontes problemáticas
 2. porto de origem/destino — fácil de fazer "profile" e evitar ataques
 3. flag bits
 4. saída/entrada (egress/ingress)
- rápido; simples; transparente para utilizadores
- sem conceito de estado; vulnerável a ataques em bugs TCP/IP
não consegue ver conexões TCP; desconhece contexto e dados aplicacionais

Ataques:

1. "Spoofing" de Endereços IP
2. Ataques da Rota de Origem
3. Ataques de Fragmentos Pequenos

Filho de Pacotes "Stateful": camada de transporte - lembra-se de pacotes TCP/UDP

EVITA SCAN TCP ACK

- consegue fazer tudo o que um filho de pacotes consegue
- regista conexões de enlace
- baseia-se em lógica protocolar para detectar maus comportamentos
- não consegue ver dados aplicacionais
- mais lento do que filho de pacotes

Proxy Aplicacional: camada aplicacional - mediador que atua em nome da outra

- missão completa das conexões e dados aplicacionais
- filtra maus dados na camada aplicacional
- pior desempenho
- cada aplicação deve ter o código do proxy associado

Política Permissiva: permitir por defeito - bloquear alguns - pior segurança

Política Restritiva: bloquear por defeito - permitir alguns - pior disponibilidade

Ordem das Regras: aplicar a primeira correspondência da lista de regras OU
aplicar a melhor correspondência para o pacote

Localização do Firewall: firewalls só podem filtrar tráfego que a atravessa

Zona Desmilitarizada: usada para servidores que requerem acesso (seletivo) a partir de dentro e de fora da firewall

Onde localizar firewalls?

Bastion Hosts: ponto crítico na rede — gateways do anfitrião

- Corre um SO seguro, nó com serviços essenciais
- Pode requerer autenticação do utilizador para aceder ao proxy ou anfitrião
- Cada proxy pode restringir funcionalidades
- Proxies pequenos, simples e seguros
- Uso limitado do disco e código só de leitura

Host-Based Firewalls: usadas para a segurança de um único anfitrião (normalmente servidor), disponíveis como add-on do SO — filtram fluxos de pacotes

- regras de filtragem afinadas, para as necessidades específicas do anfitrião
- proteção contra ataques internos e externos
- camada de proteção adicional à firewall da organização

Firewalls Pessoais: controlam o fluxo de tráfego de/para o PC — num módulo de software num PC ou num router/gateway DSL

- menos complexo
- o objetivo principal é detectar acessos não autorizados
- pode monitorizar tráfego de saída para detectar/bloquear atividade maliciosa

Firewall IV: ferramenta de scan de portos abertos através da firewall

1. Defini $TTL = 1 + \text{número de saltos para a firewall}$
 2. Definir porta de destino = N
 3. Se a firewall permitiu o porto de dados N, recebe-se "TIME EXCEEDED"
 4. Senão, não há resposta
- Não é viável através de um proxy aplicacional, porque cria um novo pacote e reescreve o antigo TTL

IPTables

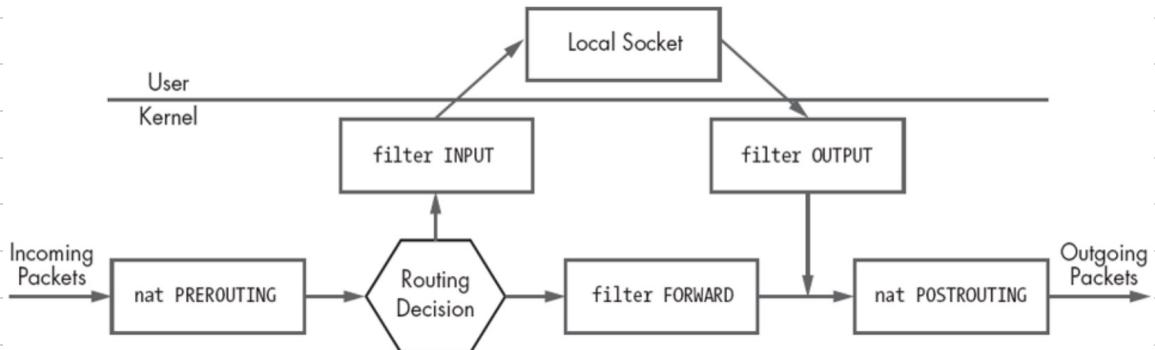
Tabela: contexto de aplicação das regras — filtro; NAT; "mangle" (modificação)

Cadeia: lugar/estado de processamento de pacotes — input; output; encaminhamento

Alvo: destino a atribuir ao pacote — Drop; Accept; Reject; Log; Return; Queue

Método:

1. Adicionar regras às tabelas, especificando os cadeias existentes
2. Quando um pacote corresponde a uma regra, o seu alvo é selecionado
 - a. Tabela de Filtro — Drop; Accept
 - b. Tabela NAT — DNAT; SNAT; Masquerade; Redirect
3. Novas cadeias podem ser criadas pelo utilizador



Target	Purpose
DROP	Discard a packet without notification to source
ACCEPT	Accept packet
REJECT	Reject packet with notification to source
LOG	Log information about the packet
RETURN	Stops evaluation of rules in the current chain
QUEUE	Puts the packet in queue to be sent to an application

Classes de Intrusos

Cibercriminosos: indivíduos ou membros de um grupo de crime organizado, com o objetivo de recompensa financeira — fóruns subterrâneos e ataques coordenados

Organizações Patrocinadas pelo Estado: grupos de hackers patrocinados por governos para conduzir atividades de espionagem ou sabotagem, com natureza coberta e persistência durante longos períodos — "Advanced Persistent Threats"

Activistas: indivíduos ("insiders" ou membros de um grupo grande) motivados por causas sociais ou políticas, cujo objetivo é promover e publicitar a sua causa, ainda que com poucas capacidades — "Hacktivists"

Outros: hackers clássicos ou "crackers" motivados pelo desafio técnico ou estímulo dos pares ou do grupo e reputação, que descobrem vulnerabilidades — "hobby"

Invisíveis: empregados com acesso aos e conhecimento dos sistemas, motivados por vingança, término de emprego ou roubo de dados de clientes para concorrentes

Capacidades:

1. **Aprendiz** — mínimas capacidades técnicas, que usam ferramentas já existentes
2. **Journeymen** — modificam ou estendem ferramentas para usar vulnerabilidades novas
3. **Mestres** — descobrem novas categorias de vulnerabilidades e criam ferramentas

Comportamento:

1. definição do alvo e recolha de informação
2. acesso inicial
3. escalamento de privilégios
4. recolha de informação ou exploração dos sistemas
5. manutenção do acesso
6. cobertura do rasto

IDS

Requisitos:

1. Disponibilidade: correr continuamente e degradar-se graciosamente
2. Segurança: tolerância a falhas e resistência a subversões
3. Desempenho: impõe um "overhead" mínimo e escalar bem
4. Adaptabilidade: configurações, adaptação a alterações e reconfiguração dinâmica

Componentes:

1. Sensores: colectar dados
2. Analizador: avaliar intrusões
3. Interface do Utilizador: ver output da comportamento do sistema de controlo

Host-Based IDS: dedicado a uma máquina ou serviços específicos - monitoriza as características de um único "host" para atividade suspeita

- Monitoriza actividades nos anfitriões por ataques conhecidos e comportamento suspeito
 - Projeto para detetar ataques como "buffer overflow" e escalamento de privilégios
 - Pode detectar intrusões externas e internas
 - Pouca ou nenhuma visão de actividades de rede
- Ex: OSSec; Tripwire; AIDE

Network-Based IDS: monitoriza tráfego de rede - analisa dados de transporte ou aplicacionais para identificar atividade suspeita

- Monitoriza actividade em pontos seleccionados da rede para ataques conhecidos
 - Examina protocolos de rede ao nível de transporte e de aplicação
 - Projeto para detectar ataques como DoS, ameaças de rede e pacotes mal-formados
 - Alguma sobreposição com firewall
 - Pouca ou nenhuma visão de ataques baseados no anfitrião
- Ex: Snort; Bro/Zeek; Suricata

FDS Híbrido/Distribuído: combina informação de múltiplos dispositivos (anfítuas e rede) combinados num analisador central

Detecção de Anomalias: conjunto de padrões de dados maliciosos ou regras de ataques — não consegue identificar ataques conhecidos

- ↪ simples; detecta ameaças conhecidas/comuns; exato/preciso; eficiente
- ↪ muitas anomalias para manter e atualizar constantemente
- ↪ não consegue detectar ataques conhecidos, variações inesperadas evitam deteção

Detecção de Anomalias: envolve a recolha de dados relacionados com o comportamento de utilizadores legítimos durante um período de tempo — o comportamento observado é analisado para determinar se corresponde a um utilizador legítimo ou a intruso através de "machine learning" e reconhecimento de padrões

Objetivo: detectar novos ataques automaticamente — deteção baseada em comportamentos

1. Estabelecer "comportamento base (normal)"
2. Definir o "threshold" de comportamento normal para comportamento anormal
3. Recolher novos dados estatísticos
4. Medir desvios em relação à base
5. Se o "threshold" for excedido, lançar um alarme

Abordagens de Classificação:

1. Estatística — análise do comportamento observado usando modelos univariados, multivariados ou temporais dos dados observados
 - ↪ simples; leve; assumções pequenas no comportamento conhecido
 - ↪ seleção de variáveis para análise; difícil equilibrar FP vs FN

2. Baseada em Conhecimento — usa um sistema que classifica o comportamento observado de acordo com um conjunto de regras que modelam comportamento legítimo

- ↪ robusta; flexível
- ↪ consome tempo; requer conhecimento profundo de padrões aplicacionais

3. Machine Learning — determina automaticamente um modelo de classificação adequado a partir dos dados de treino usando técnicas de mineração de dados

- ↪ classificação automática; eficiência depois de treino
- ↪ consome tempo e computação; "overfitting"

• Um sistema de intrusão estático coloca grande pressão no administrador de sistemas, mas um IDS em evolução torna possível o atacante manipular o comportamento e convencer lentamente o IDS de um padrão anormal

Falso Positivo (FP): ataque identificado quando não ocorre

Falso Negativo (FN): ataque identificado como comportamento adequado

Falácia do Rácio-Base: probabilidade de um evento condicional avaliada sem considerar o "rácio-base" desse evento

Abordagens de Resposta de IDS:

1. Bloqueio Preemptivo: previne intrusões antes de ocorrerem, bloqueando endereços IP — risco de bloquear utilizações legítimas ou de o atacante mudar de máquina

2. Infiltração: infiltrar-se em grupos online de hacking

3. Disfarce de Intrusos: tornar o sistema um alvo menos apelativo (esconder os ativos valiosos) e fazê-lo parecer mais seguro do que é (ter avisos de monitorização)

4. Defleção de Intrusos: configurar um sistema ativo, mas falso, com o propósito de monitorizar/compreender a atividade do atacante e os padrões de intrusão

Honey Pot: sistema projetado para atrair um potencial atacante para longe de sistemas críticos, recolhendo dados sobre a sua atividade e encorajando-o a permanecer no sistema

Domínios de Confiança

- A segurança é estabelecida de acordo com regras específicas
- O paradigma cliente-servidor assume implicitamente que ambas as partes não confiáveis
- Os protocolos de segurança não interagem com o hardware

Âncora de Confiança: peça central de segurança, segura por assinatura, usada para armazenar e processar material criptográfico

- compacto "by design"
- curva acelerada de aprendizagem
- limitado computacionalmente
- manutenção/implementação cuidadosa

Smart Card: armazena e processa informação num micro-controlador embutido em plástico - portátil, resistente à fraude e sem necessidade de recursos externos

- 2FA
- PKI
- económico
- personalizável

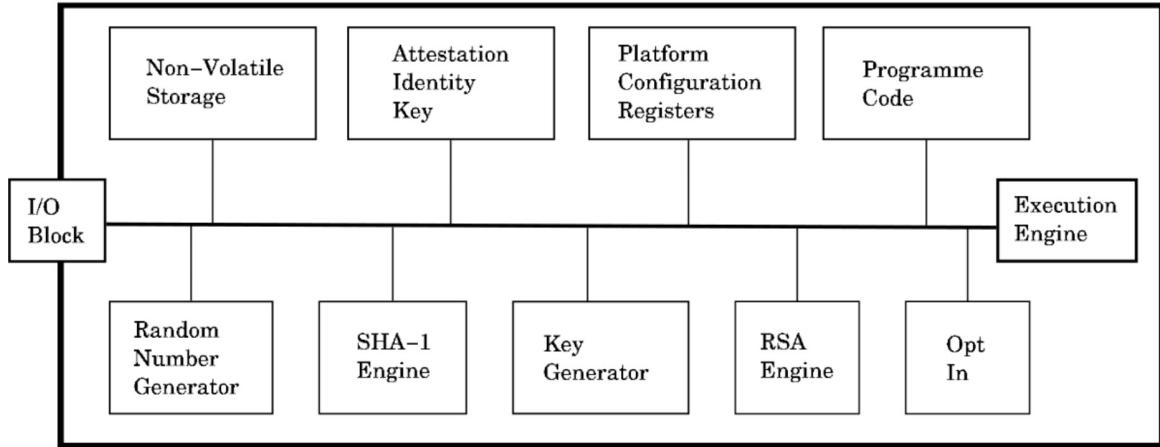
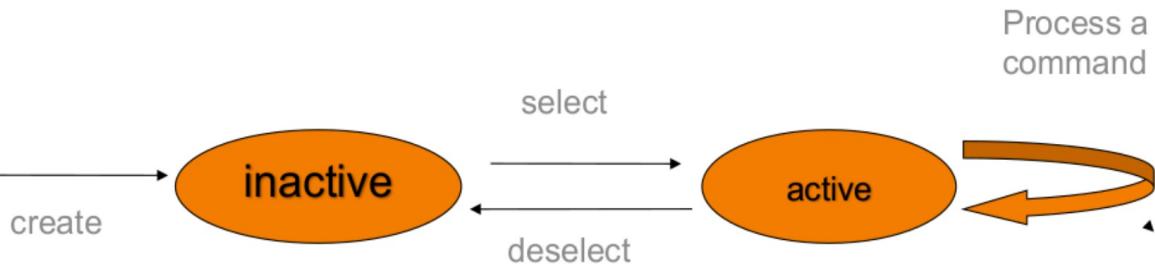
Java Card: portátil; seguro; independência de hardware; baixo-custo

Applet: programa em Java que adere a um conjunto de convenções para ser executado no ambiente de execução do Java Card, podendo ser dinamicamente carregado no cartão, armazenar dados e realizar operações massivas, com isolamento garantido pelo ambiente - podem coexistir múltiplos "applets" no mesmo cartão

APDU: pacotes de dados do protocolo de comunicação ao nível da aplicação, escondido dos programadores e "half-duplex" (comunicação uni-direcional de cada vez)

Trusted Platform Module (TPM): microchip para operações de segurança que ~~não~~ ^{não} assiste no sistema operativo para processamento, realizando operações criptográficas isoladas

"Root of Trust for Measurement" (RTM): valor predefinido para código crítico, verificado contra o armazenado no TPM - ponto de partida na cadeia de medições



22/23

1.1. Um ataque à integridade do sistema visa alterar/manipular o sistema/programa e/ou a informação de forma não autorizada, para que não realize a sua função pretendida de forma intacta. Um ataque à disponibilidade do sistema visa negar ou dificultar o acesso ao serviço a utilizadores autorizados. Assim, a diferença é que o primeiro afeta o funcionamento do sistema, enquanto o segundo torna o sistema inoperacional.

1.2. Uma ferramenta para proteger a confidencialidade dos dados de um sistema é a encriptação. A encriptação funciona ao receber uma mensagem e uma chave para produzir um criptograma através de determinados algoritmos/procedimentos, para que (idealmente) o criptograma não revele/liberte nenhuma informação sobre a mensagem, a não ser para os detentores da chave, que podem recuperar a mensagem original.

1.3. O Carlos pode interceptar o envio da chave pública da Alice para o Bob e de Bob para a Alice, substituindo ambas pela chave pública gerada pelo Carlos. Assim, quando a Alice e o Bob utilizarem a chave pública recebida para cifrar informação, estariam, na verdade, a usar a chave pública do Carlos, pelo que o Carlos pode interceptar a informação cifrada e decifrá-la usando a sua chave secreta, quebrando a confidencialidade das mensagens trocadas. O Carlos pode ainda cifrar as mensagens decifradas com a chave pública do destinatário original e transmiti-las, dissimulando a sua presença na rede de comunicação.

2.1. Autenticação é o processo para determinar se deve ser permitido acesso de um utilizador a um sistema. Autorização é o processo de decidir que ações um utilizador pode realizar no sistema. Assim, o primeiro visa gerir os acessos a um sistema, enquanto o segundo visa permitir ou proibir as ações do utilizador (autenticado) dentro do sistema.

2.2.

a) Carlos consegue efectuar um ataque de repetição para se autenticar como Alice. Para tal, basta enviar as mensagens observadas enviadas pela Alice na troca legítima de autenticação, fazendo-o no mesmo minuto (do mesmo ou de outro dia) em que a Alice o faz, não precisando, assim, de conhecer o valor da sua password.

b) Uma correção possível para este protocolo é substituir o uso do minuto do dia pelo uso de um número de uso único ("nonce") como o valor de um contador ou o número de milissegundos desde 1/1/1970, impedindo ataques de repetição por ser irrepetível.

3.1. O papel do protocolo handshake do TLS é estabelecer os parâmetros criptográficos a utilizar na comunicação. Assim, o protocolo é usado antes de quaisquer dados aplicacionaisarem transmitidos, permitindo autenticação mutua e negociação dos algoritmos de encriptação, de MAC e de chaves criptográficas.

3.2. A diferenciação entre conexão TLS e sessão TLS permite minimizar/baixar os custos da fase de negociação do protocolo TLS, executada pelo protocolo handshake. Assim, como uma sessão pode ter várias conexões, alguns parâmetros criptográficos como os algoritmos de encriptação e MAC a utilizar e a chave secreta simétrica neste podem ser definidos para a sessão, enquanto os parâmetros criptográficos efêmeros de curta duração como as chaves de encriptação e de MAC não são definidos ao nível da conexão, partindo tempo e recursos computacionais.

3.3. A afirmação é falsa porque o TLS opera entre a camada de transporte e de aplicação, portanto acima (e não abaixo) da camada de rede. Assim, o TLS não acompanha o processo de roteamento IP porque este ocorre na camada de rede, enquanto o TLS opera acima da camada de transporte, na camada aplicacional.

3.4. Dois métodos possíveis de autenticação no SSH são: (1) chave pública e (2) palavra-passe. No método de chave pública, o cliente envia uma mensagem para o servidor com a sua própria chave pública e armada com a chave privada - o servidor verifica se a chave é aceitável para autenticação e se a armadura está correta. No método de palavra-passe, o cliente envia uma mensagem contendo a sua palavra-passe, encriptada pelo Protocolo da Camada de Transporte - o servidor verifica-a.

3.5. O IPSec pode ser mais adequado do que o TLS para em Virtual Private Networks (VPNs). Como o IPSec opera na camada de rede, protege todo o tráfego IP entre as redes, independentemente da aplicação, pelo que permite a execução segura de diferentes protocolos (SSH, HTTP, FTP, ...) de forma transversal.

3.6.

a) Em modo transporte, o IPSec adiciona informação/segurança ao pacote original, pelo que autentica (AH) ou encripta (ESP) a componente "Data" e alguns campos do "IP Header" (estáticos). Isto é vantajoso para comunicação host-to-host porque é extremamente eficiente, mas o atacante pode ver partes do cabeçalho.

b) Em modo túnel, o IPSec encapsula todo o pacote ("IP Header" e "Data") num novo pacote, protegendo a totalidade do conteúdo. Isto é vantajoso para comunicação firewall-to-firewall porque o endereço IP do novo pacote (externo) para a referir-se à firewall, ocultando o endereço original de atacantes.

3.7.

- a) Wireshark NÃO, porque serve apenas para analisar comunicações/pacotes de rede
- b) Nmap SIM, porque é uma ferramenta de scan de rede que identifica dispositivos
- c) IPTables NÃO, porque é uma ferramenta de gestão de firewalls em Linux
- d) Ping SIM, porque permite testar a conectividade de uma máquina através do seu endereço, verificando se responde a pedidos de rede

3.8. ICP Idle Scan é um tipo/método de scan de redes que utiliza máquinas zumbi para enviar os pacotes. O scan consiste em fazer "spoofing" do endereço IP de origem dos pacotes para o endereço IP de máquinas zumbi e depois verificar se a máquina "zumbi" obtém resposta, indicando o estado do alvo.

4.1.

a) Um exemplo de ataque de reflexão em que Alice é a vítima e Bob o atacante é o seguinte: o Bob envia para o serviço de eco no porto 7 pacotes com o conteúdo de CharGen com endereço IP de origem "spoofed" como o da Alice, pelo que a Alice vai ser inundada de pacotes (Echo-CharGen). OU o Bob envia para "broadcast" um pacote com endereço IP de origem "spoofed" como a da Alice, pelo que a Alice vai ser inundada com pacotes de eco de muitas máquinas da rede (Smurf).

b) Por exemplo, para prevenir os dois ataques de reflexão anteriores podem ser implementados desativar os serviços de eco, de CharGen (Echo-CharGen) e de "broadcast" se possível. Podem também ser implementados mecanismos anti "spoofing" (verificação do endereço IP de origem) e limitar a taxa de respostas UDP.

4.2. Num ataque de "Denial-of-Service" ganha quem tem mais largura de banda. Portanto, se o atacante tiver mais largura de banda do que a vítima consegue sobrecarregar a ligação de pacotes e comprometer a capacidade da vítima em receber outras comunicações mas, se a vítima tiver mais largura de banda, consegue responder a todos os pacotes enviados pelo atacante sem comprometer a sua disponibilidade.

4.3. A computação na nuvem e a utilização de software-as-a-service aumenta a superfície de ataque dos sistemas porque expõe serviços das organizações através da internet, em ambientes que podem ser vulneráveis e que, ao aumentarem as dependências e serviços abertos disponíveis, aumentam os riscos.

5.1. Firewalls são um mecanismo importante na proteção contra ataques de DoS porque decidem o que deixar entrar na rede interna, controlando os acessos à rede com múltiplos níveis de granularidade, pelo que conseguem filtrar determinados tipos de pacotes com base no seu endereço IP de origem e protocolo, por exemplo, bloqueando-os e impedindo que cheguem ao sistema de maneira a não comprometer a sua disponibilidade.

5.2. Ter uma firewall como "application proxy" em vez de ao nível da camada de rede tem como vantagem ter uma visão completa das conexões e dados aplicacionais e filtros maiores dados na camada aplicacional, mas tem como desvantagem ter um pior desempenho e cada aplicação deve ter o código do proxy associado.

5.3. Um ataque de port scanning via TCP consiste em: (1) o atacante envia um pacote SYN à vítima e (2) se a vítima responder com SYN-ACK, então esse porto está aberto para conexões TCP. O atacante pode responder com RST para terminar a conexão.

5.4. Mecanismos de firewall e sistemas de deteção de intrusões (IDS) podem complementar-se porque a firewall tenta controlar os acessos à rede através da inspeção e filtro de pacotes que a atravessam, enquanto o IDS monitoriza os pacotes trocados na rede interna à procura de comportamentos suspeitos ou assinaturas que indiciem a presença de um atacante na rede que terá eventualmente escapado à firewall

5.5. A configuração de IDS mais adequada é "network-based" porque todos os participantes da rede acedem ou disponibilizam serviços internos e externos, pelo que o mais importante é monitorizar todo a rede. Se P2, P3 e P4 só fizessem acessos a P1 e não a outros serviços na internet, a solução mais adequada seria um "host-based" IDS em P1.

5.6. É importante minimizar o número de falsos positivos nos IDS porque são identificações de ataques que não ocorrem que vão sobrecarregar a equipa de segurança, deslocando o seu foco de eventuais verdadeiros positivos que devem ser investigados para garantir a segurança da organização.

TAKE 2

1.1. Dois mecanismos que ajudam a proteger a disponibilidade do sistema são: (1) proteção física - infraestrutura que pode manter informação disponível mesmo no caso de desafios físicos e (2) redundância computacional - múltiplos servidores e "backends" que podem garantir que o serviço se mantém disponível mesmo no caso de (algumas) falhas.

1.2. Um ataque do tipo "Man-in-the-Middle" consiste em interceptar um fluxo de dados, (às vezes) modificá-lo e retransmiti-lo, comprometendo a confidencialidade e integridade do sistema.

1.3. As infraestruturas de chave pública protegem contra ataques de "Man-in-the-Middle" ao utilizar certificados digitais armazenados por Autoridades Certificadoras que aseguram a identidade do sujeito da certificação ao armazená-la com a sua chave secreta, sendo verificável com a sua chave pública, garantido que a comunicação é efectuada com o destinatário correto e não com um atacante, de acordo o nível/grau de confiança do remetente na CA.

2.1. "Freshness" (frescura) ao falar de "tokens" significa que um determinado valor gerado tem alta probabilidade de ser único, estando, portanto, protegido contra ataques de repetição, por nunca ter sido gerado previamente.

2.2.

a) Um problema central desse protocolo é que a chave criptográfica gerada pelo Bob que não é usada para cifrar informações não é encriptada, mas apenas armazenada, pelo que é visível para qualquer atacante, quebrando a confidencialidade da comunicação futura encriptada com essa chave.

b) Uma correção possível é o Bob armazenar o "nonce" e encriptar a chave K (com a chave partilhada anteriormente), para que a Alice consiga verificar a identidade do Bob através da armazenatura do "nonce" e obter uma chave de cifra de forma segura ao desencriptar "K" usando a chave previamente partilhada, garantindo a confidencialidade de comunicações futuras.

3.1. A frase é verdadeira porque, para além da encriptação do TLS funcionar como defesa em profundidade para os dados já cifrados dado que um atacante teria de quebrar duas chaves para comprometer a integridade do sistema, o TLS ainda fornece autenticação e integridade aos dados trocados, aumentando a sua segurança.

3.2. O protocolo de alerta do TLS tem como papel provocar avisos ou terminar conexões através do envio de uma mensagem comprimida e encriptada de dois bytes em que o primeiro se refere à severidade do alerta (fatal, termina a conexão imediatamente; não-fatal, outras conexões para a mesma sessão podem continuar, mas não podem ser estabelecidas conexões adicionais) e o segundo específico, de maneira a terminar conexões potencialmente insecuras.

3.3. Duas desvantagens da utilização do TLS são: (1) "overhead" computacional devido ao aumento dos custos e do tempo de processamento necessário para trocar informação, principalmente pelo protocolo "handshake" e (2) possíveis falhas de configuração devida a complexidade do protocolo, como escolher algoritmos ou parâmetros inseguros, que pode abrir falhas de segurança fáceis de detectar, comprometendo o sistema.

3.4. O mecanismo de "Port Forward" do SSH fornece a capacidade de converter qualquer conexão TCP insegura através de "tunneling" SSH, isto é, ao fazer encaminhamento do tráfego de rede de um porto local ou remoto para outro porto, protegendo os dados pelo protocolo SSH e criando túneis de comunicação seguros.

3.5. IPsec é um protocolo que refina o protocolo IP, fornecendo segurança na camada de rede de forma transparente às aplicações, mas cria conflitos com algumas firewalls. TLS é middleware entre a aplicação e TCP, sendo um sistema de propósito-geral implementado como um conjunto de protocolos que assentam no TCP para assegurar garantias de entrega de mensagens. A diferença entre utilizar ambos é que o primeiro é implementado na camada de rede de forma transparente/tunelar às aplicações, enquanto o segundo é implementado entre a camada de transporte e a camada aplicacional.

3.6. Security Associations (SA) são estabelecidas entre duas entidades com os algoritmos e parâmetros criptográficos a utilizar na comunicação entre elas. Security Policy Databases (SPD) são os meios pelos quais o tráfego IP se relaciona com SAs, ou seja, tabelas cujas entradas definem subconjuntos de tráfego IP e apontam para SAs. Assim, a relação é que SPD definem o que fazer com o tráfego e SA definem "como o fazer", garantindo segurança.

3.7.

- a) Wireshark, NÃO porque apenas serve para inspecionar pacotes do tráfego na rede
- b) NMap, SIM porque permite fazer "scans" de portos e assim identificar serviços disponíveis
- c) IPTables, NÃO porque não gerenciam as regras de gestão de firewalls em Linux
- d) Ping, NÃO porque apenas permite testar a conectividade de uma máquina

3.8. A ferramenta NMAP é útil no contexto de testes de penetração porque permite fazer "scans" de endereços IP, de portas e de serviços para descobrir hosts acessíveis com portas abertas e serviços suspeitos, eventualmente vulneráveis.

4.1.

a) Um sistema "Zombie" é uma máquina comprometida na qual o atacante usa uma falha no sistema operativo ou numa aplicação comum para ganhar acesso e a conseguir controlar remotamente. Por exemplo, pode ser criado através de uma vulnerabilidade do Microsoft Windows que permite execução remota de código.

b) Uma "Botnet" é uma rede de computadores infectados com "malware" malicioso ("malware") que permite que sejam controlados por atacante, gerando facilmente enormes volumes de tráfego de diferentes origem que podem sobrecarregar um servidor ao ponto de comprometer a sua disponibilidade - Denial-of-Service.

4.2. Um DNS Amplification Attack funciona ao usar pedidos DNS com endereço IP de origem "spoofed" como sendo o alvo, explorando o comportamento do DNS (com o argumento "ANY") para converter um pedido pequeno (60 bytes) numa resposta muito maior (512-4000 bytes) - o atacante envia pedidos múltiplos para servidores bem conectados, inundando o alvo de forma eficaz e difícil de detectar. As mitigações possíveis são: (1) reduzir o número total de resoluções DNS abertas, (2) restringir um resoluedor DNS para não responder a "queries" de fontes confiáveis e (3) ter os ISPs a detectar ativamente endereços IP "spoofed".

4.3. Platform-as-a-Service é um modelo de computação na nuvem que fornece uma plataforma e ambiente para desenvolver, testar, implementar e gerir aplicações sem a complexidade de manter a infraestrutura subjacente. Uma vulnerabilidade em PaaS que não existe quando o servidor é uma máquina dedicada pode ser por exemplo fugas de dados entre diferentes "tenants" resultantes de um mau isolamento/regras de segurança entre elas.

5.1. Um critério de filtragem que permite a firewalls protegerem sistemas de ataques por denial-of-service pode ser por exemplo limitar a taxa de pacotes ou bloquear dependendo do IP de origem (victim) ou do protocolo utilizado.

5.2. 5.2.

5.3. Defesa em Profundidade é importante para que, se um atacante conseguir quebrar ou contornar um mecanismo de defesa, ainda tenha que quebrar ou contornar outros mecanismos de defesa para comprometer o sistema, dificultando o ataque.

5.4. 5.4.

5.5. Um "Host-Based IDS" é dedicado a uma máquina ou único específico, monitorando as características de um único "host" para atividade suspeita. Um "Network-Based IDS" monitora tráfego de rede, analisando dados de transporte/aplicação para identificar atividade suspeita. Assim, a diferença é que o primeiro visa detectar intrusões num único ambiente, enquanto o segundo procura intrusões em toda a rede.

5.6. A abordagem de classificação mais adequada é estatística porque as mensagens têm um comportamento bem definido e conhecido, pelo que não é necessário introduzir o risco de um atacante conseguir manipular o comportamento e comutar lentamente o modelo de "machine learning" de um padrão anormal.

6.1. Hardware confiável como "Trusted Platform Module" (TPM) pode armazenar de forma segura os parâmetros criptográficos (chaves) de forma segura contra ataques.

6.2. Um ataque de timing consiste em extraír informações sobre o funcionamento do hardware confiável através dos diferentes tempos que demoram as suas computações.