Contramedidas e hardening

Prof. Frederico Sauer Guimarães Oliveira

Descrição

As contramedidas de segurança e o hardening de equipamentos e sistemas na proteção contra os ataques cibernéticos.

Propósito

O cenário atual da economia mundial, fortemente baseada em operações com o uso de sistemas computacionais, demanda proteções contra os ataques cibernéticos que podem causar impactos significativos. Com o aumento das formas e da intensidade dos ataques, é importante que todo profissional de TI possua competências na definição de contramedidas e hardening para contribuir com a proteção das informações de uma corporação.

Objetivos

Módulo 1

Ferramentas de segurança e de criptografia

Relacionar ferramentas de segurança e de criptografia com objetivos mitigatórios.

Módulo 2

Protocolos de comunicação IP, TCP, UDP, DNS e HTTP

Identificar vulnerabilidades e recursos de segurança no contexto dos protocolos de comunicação IP, TCP, UDP, DNS e HTTP.

Módulo 3

Hardening de segurança em ambientes Linux e Windows

Selecionar recursos para hardening de segurança em ambientes Linux e Windows.

Módulo 4

Estratégias de segurança para redes sem fio (wireless) e IoT

Selecionar estratégias de segurança para redes sem fio (wireless) e IoT.

Introdução

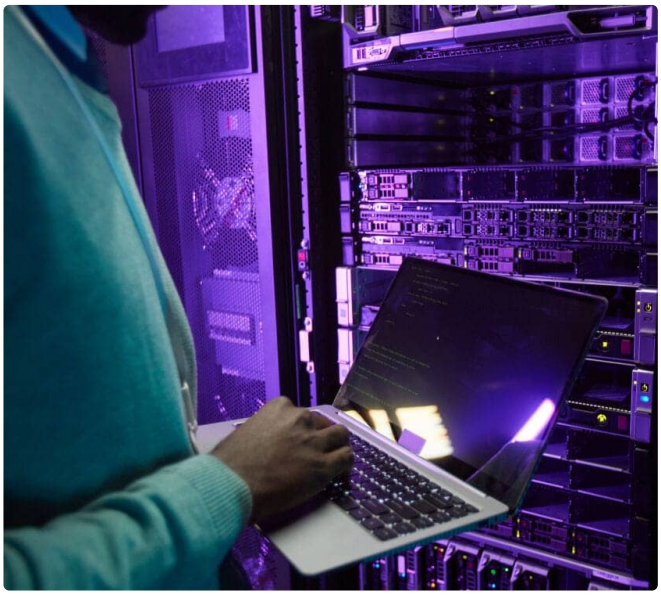
A sociedade moderna incorporou recursos computacionais a atividades cotidianas. Pessoas e empresas migraram suas informações para mídias digitais, tornando-nos dependentes de dispositivos como celulares, computadores, conectividade pública e privada e, mais recentemente, armazenamento em nuvem.

Eletrodomésticos, lâmpadas, sensores e outros foram integrados às redes, no fenômeno chamado Internet das coisas (IoT). Usamos esses recursos para aprender e ensinar, para nos relacionar com outras pessoas e nos divertir, mas também usamos para comprar, movimentar recursos financeiros e outras atividades que envolvem transferência digital da propriedade de bens e valores. Com isso, surgiram os criminosos virtuais.

É importante frisar que a internet foi idealizada por estudantes e pesquisadores, cuja principal preocupação era a de garantir a comunicação, mesmo na indisponibilidade de elementos funcionais de uma rota, buscando percursos alternativos. Desse modo, nossos “pacotes” navegam entre a origem e o destino por caminhos variáveis e imprecisos, facilitando os ataques cibernéticos.

Questões de segurança não foram incluídas nas especificações, ou eram muito fáceis de serem contornadas, então, vários protocolos de comunicação em uso atualmente possuem as chamadas brechas de segurança. Aplicações nem sempre são desenvolvidas acompanhadas de um framework específico para serem seguras. Equipamentos são instalados sem as devidas configurações seguras.

Existem medidas de proteção contra os ataques cibernéticos, mas elas precisam ser aplicadas de acordo com os riscos existentes e as boas práticas de segurança. Neste estudo, vamos entender a aplicabilidade das medidas de proteção disponíveis, caso a caso.



1 - Ferramentas de segurança e de criptografia

Ao final deste módulo, você será capaz de relacionar ferramentas de segurança e de criptografia com objetivos mitigatórios.

O que são as ferramentas?

As ferramentas de segurança e a criptografia são recursos essenciais na mitigação de incidentes de segurança. O uso desses recursos depende de planejamento e conhecimento do universo de vulnerabilidades que o

ambiente possui, de modo a escolher a **solução certa** para cada problema. Como as ferramentas de segurança usam recursos criptológicos, vamos iniciar este módulo pela criptografia.

Criptografia

Há registros que, desde os primórdios da história, pessoas usam artifícios para disfarçar informações sensíveis (STALLINGS, 2015). Com os conflitos mundiais, e em especial a Segunda Guerra Mundial, a criptografia se tornou popular e passou a ser incorporada aos diversos tipos de sistemas.

> **O objetivo principal da criptografia é garantir a confidencialidade da informação.**

Diversas bibliotecas são públicas e é trivial agregar a criptografia no desenvolvimento de aplicações e protocolos de comunicação. É vital que se crie a cultura de introduzir a segurança como um requisito das aplicações durante o processo de levantamento de requisitos e o desenvolvimento em si.

A criptografia pode ser dividida em dois grandes tipos, de acordo com o uso das chaves: **criptografia simétrica** e **assimétrica**. Vamos detalhar as duas.

Criptografia simétrica

Na criptografia simétrica, um par de usuários que desejam se comunicar com privacidade usam apenas uma chave, a **chave secreta**. Ela precisa ser combinada **antes** do início da transmissão das informações, com risco de comprometimento. Na famosa cifra de César (100 a.C. – 44 a.C.), caracteres eram substituídos por outros com um simples deslocamento da posição do caractere desejado (ANDERSON, 2017).

Com chave = 3 (um deslocamento de 3 posições), teremos a seguinte correspondência:

Letra Desejada	A	B	C	D	E	F	G	H	I
Letra cifrada	D	E	F	G	H	I	J	K	L

Comentário

A palavra ATACAR, por exemplo, ficaria DWDFDU, portanto, ilegível. É claro que uma simples análise estatística de um criptograma mais longo permitiria a quebra da chave, ou seja, do número de posições deslocadas.

Algoritmos simétricos são rápidos, em comparação com os assimétricos, devido ao baixo custo computacional de suas operações típicas, que são:

- **Substituições**: Bits são trocados por outros.
- **Permutações**: A posição dos bits é trocada.
- **Deslocamentos**: Como em uma lista circular, os bits são deslocados de suas posições mantendo a ordem entre eles.

Sabendo disso, é importante considerar que há dois tipos de algoritmos simétricos:

Cifras de bloco

Os bits são encriptados em grupos de tamanho fixo. Reduz a velocidade do fluxo, mas alcança grande entropia nos criptogramas.

Cifras de fluxo

O algoritmo cifra de um bit a alguns bytes por vez, durante a transmissão. Apesar de rápidos, é complexo alcançar alta entropia.

Em 1997, em virtude da obsolescência do principal algoritmo simétrico de bloco da época, o DES, o National Institute of Standards and Technology (NIST) fez um concurso para substituí-lo, com os requisitos (STALLINGS, 2015):

Código livre e público. A robustez não poderia depender do segredo do código, e todos poderiam usar livremente;

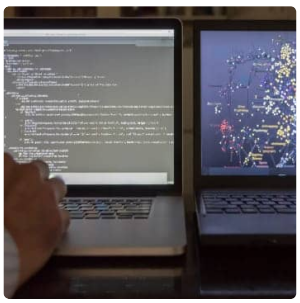
Baixo custo computacional, e implementável em hardware.

Nesse concurso, saiu-se vitorioso o algoritmo Rijndael, que passou a ser chamado de AES. Suas principais características são:

Chaves secretas de 128, 192 ou 256 bits. Apesar de padronizado em 1999, a maioria das soluções ainda usa chaves de 128 bits com grande robustez;

Blocos de 128 bits, apesar da especificação original do Rijndael permitir blocos de 128, 192 ou 256 bits;

Rápido e com baixos requisitos de memória, quando comparado com os outros.



Algoritmos simétricos são sujeitos à análise estatística, como no exemplo da cifra de César. Um texto claro criptografado com a mesma chave gerará o mesmo criptograma. Para evitar essa facilidade que possibilita a quebra da chave, são usados os **modos de cifra**, como o Cypher Block Chaining (CBC) ou o Counter (CTR). Esses modos promovem manipulações durante cada encriptação de bloco que eliminam a regularidade nos processos de cifragem, aumentando ainda mais a entropia resultante.

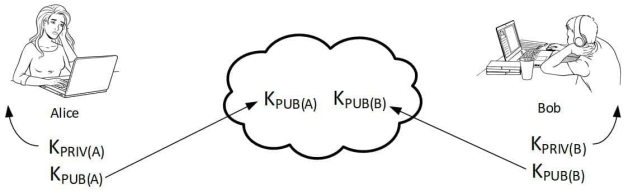
> **Um ponto de fragilidade e risco no emprego da criptografia simétrica é a demanda da combinação da chave secreta entre os pares.**

A criptografia assimétrica resolve isso e acrescenta outras funcionalidades, como veremos a seguir.

Criptografia assimétrica

A criptografia simétrica é rápida, eficaz, consome poucos recursos e pode ser implementada em hardware. Mas a demanda de troca da chave secreta por um canal inseguro é um problema real. Para resolver isso e agregar outras funções, existe a criptografia assimétrica.

Nela, são usadas duas chaves que possuem propriedades mútuas. Tudo que for encriptado com uma delas, apenas poderá ser decifrado pela outra, do mesmo par. Para que isso aconteça, algoritmos assimétricos são dependentes de operações complexas, de alto custo computacional, em comparação com a simétrica. Usam chaves tipicamente maiores que nos algoritmos simétricos, mas pesquisas vêm conseguindo bons resultados na redução do custo e do tamanho das chaves (SAADI; KUMAR, 2020). Esse par de chaves tem propriedades peculiares e de grande interesse para a segurança nas comunicações globais. A imagem a seguir ilustra essas propriedades.

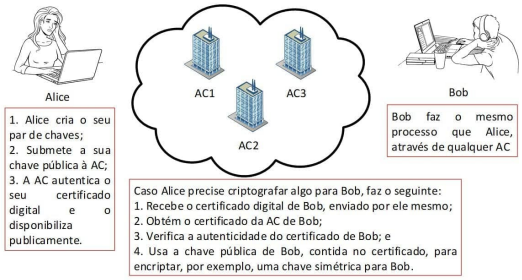


Processo de autenticação e integridade da mensagem.

Alice (A) e Bob (B) criam o seu par de chaves. Enquanto a chave **pública** é disponibilizada em servidores de chaves, a **privada** é guardada de forma segura, individual e privativa de seu proprietário, por exemplo, armazenada criptografada em um token.

Para Alice enviar uma mensagem (MSG) de forma sigilosa para Bob, ela precisa usar a chave pública de Bob. Ela busca essa chave no servidor e verifica a sua autenticidade. Alice usa um algoritmo assimétrico para encriptar MSG com a chave pública de Bob e envia a ele de forma confidencial. Isso é representado da forma $\mathrm{K_{PUB(B)}}\,(\mathrm{MSG})$, que significa "MSG encriptada com a chave $\mathrm{K_{PUB(B)}}$". Bob precisa usar a sua chave privada para desfazer esse procedimento, e só Bob a possui.

Agora, suponha que Alice deseje autenticar MSG. Para isso, ela precisa usar a sua própria chave privada. É claro que qualquer um pode decifrar esse criptograma, uma vez que a chave necessária - $\mathrm{K_{PUB(A)}}$ - está disponível publicamente, mas isso não é um problema, já que o objetivo aqui é a autenticidade de MSG. Esse mecanismo depende de um terceiro confiável, que autentique as chaves públicas dos usuários. Para isso, existe uma infraestrutura para armazenamento e distribuição de chaves públicas em certificados digitais padrão X.509. A figura a seguir sintetiza o processo de criação e distribuição de chaves usando a Public Key Infrastructure (PKI).



Uso da PKI.

Como ilustrado na figura, qualquer usuário do mundo pode criar e usar certificados digitais. Para garantia de autenticidade desse documento digital são usadas as Autoridades Certificadoras (AC), que usam suas próprias chaves privadas para isso.

Conforme vimos até aqui, a criptografia **assimétrica** oferece:

Confidencialidade

Por meio do uso da chave pública do destinatário para criptografar informações.

Autenticidade

Com a garantia dada pelas AC de que as chaves efetivamente pertencem aos donos declarados na comunicação.

Integridade

Com o uso de assinaturas digitais, que serão explicadas mais adiante.

Em termos práticos, as soluções atuais combinam o melhor arranjo, da seguinte forma:

- A criptografia do tráfego é feita de forma simétrica, para explorar a rapidez desses algoritmos, garantindo a confidencialidade do tráfego;
- A criptografia assimétrica é usada para a combinação da chave simétrica e para as assinaturas digitais, que oferecem os recursos de integridade e autenticidade.

Algoritmos de autenticação e integridade

Conforme dissemos, a criptografia assimétrica é lenta e consome muitos recursos computacionais, como memória e CPU. Além disso, como o objetivo é apenas verificar se a origem da mensagem é quem diz ser, e se a mensagem não foi alterada em trânsito, não há necessidade de se usar a mensagem inteira, mas apenas uma "assinatura" dela, ou seja, um conjunto pequeno de bits que tenha baixíssima probabilidade de coincidir com a assinatura da mesma mensagem, após alterações. Para isso, foram criadas as funções hash criptográficas.

As **funções hash criptográficas H** possuem as seguintes características (GOODRICH, 2013):

- Independentemente do arquivo de entrada MSG, é gerada por um algoritmo H uma string de bits de tamanho fixo h = H(MSG) – o hash da mensagem;
- As operações para obtenção desse hash implementadas por H são simples e rápidas, como somas de verificação e XOR;
- A função H é unidirecional, ou seja, após gerar o hash h, é computacionalmente inviável reverter o cálculo e reaver a mensagem original;

- A probabilidade de se alterar uma mensagem e obter o mesmo hash h é desprezível, dado que os algoritmos H padronizados possuem o chamado “efeito cascata”. A alteração de um único bit muda consideravelmente o hash h;

- Obter uma “colisão”, ou seja, gerar uma mensagem MSG’ cujo hash seja intencionalmente igual ao de outra mensagem MSG, é também computacionalmente inviável.

Há vários algoritmos de hash padronizados para uso livre e gratuito, como o Secure Hash Algorithm (SHA).

Ferramentas de segurança

Uma forma de organizar as ferramentas de segurança é dividi-las em dois grupos (ANDERSON, 2017): a segurança do sistema e as aplicações de autenticação, autorização e accounting (AAA).

Segurança do sistema

Sistemas computacionais envolvem o uso de várias aplicações, além do próprio sistema operacional, que podem ter vulnerabilidades. Além disso, é importante que algum mecanismo opere nas interfaces da rede e do equipamento, permitindo ou bloqueando a entrada e a saída de pacotes, de acordo com um conjunto de regras. Assim, contribuem para evitar que ações maliciosas comprometam o sistema.

Firewall

Recurso que separa um ambiente perigoso de um seguro, impedindo a passagem de mensagens de acordo com regras. Tipos de firewall (ANDERSON, 2017):

Filtros de pacotes

Regras para avaliar a entrada e a saída do perímetro da rede, com base em informações dos protocolos IP, TCP, UDP e ICMP. Exemplo de regra IPTABLES:

```
$ iptables -A INPUT -i etho -s 99.99.99.99 -j DROP
```

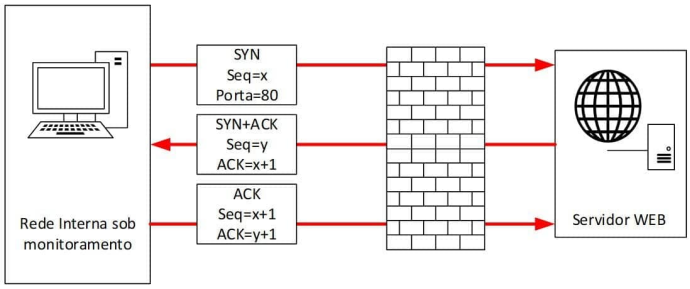
Essa regra instrui o descarte dos pacotes vindos do IP 99.99.99.99 de forma silenciosa.

Filtros de pacotes não fazem inspeção do conteúdo do tráfego, nem armazenam informações do estado de conexões TCP, de forma que são

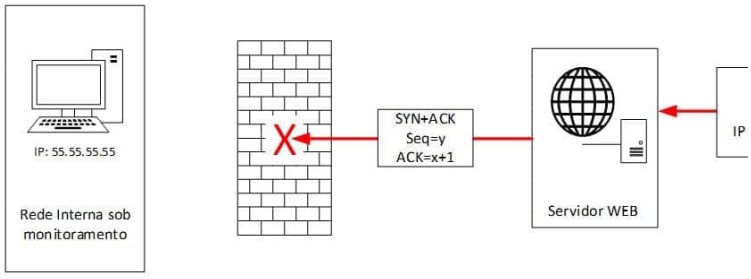
simples e com baixa sobrecarga em relação a outras soluções, mas são mais fáceis de serem burlados.

Gateways de circuito

Operam na camada de transporte, com mais opções funcionais. Em cada sessão TCP, os pacotes são remontados e avaliados antes de entrar ou sair da rede. Ataques de negação de serviço em que respostas de pedidos de conexão são enviadas sem que tenha havido o pedido não passam por esse firewall, uma vez que ele registra todos os pedidos realizados e só deixa passar respostas legítimas. As imagens a seguir ilustram a operação de um Circuit-Level GW.



Conexão normal pelo Circuit-Level Gateway.



Ataque de Negação bloqueado pelo Circuit-Level Gateway.

Na primeira situação, o host com o IP 55.55.55.55 faz um acesso a um servidor web (porta 80) e o firewall deixa a conexão ocorrer porque a lógica da conexão é seguida. Já na segunda figura, um atacante faz um spoofing usando o IP do alvo, 55.55.55.55, e submete ao servidor web. O servidor, então, responde ao IP indicado com um aceite de conexão (SYN+ACK), porém, como o firewall não tem registro dessa conexão no nível de transporte, descarta a mensagem.

Relay (Gateway) de aplicação

Componentes que inspecionam o conteúdo das mensagens da aplicação, remontando e avaliando toda a mensagem antes de permitir a sua entrada, bloqueando código malicioso em e-mails e navegação HTTP. É mais oneroso para o tráfego por precisar recuperar mensagens inteiras de toda a rede. Por isso, costumam ser usados para tipos específicos de aplicação.

Detecção de Intrusos

Podem ser implementados em software ou em hardware, na forma dos Intrusion Detection Systems (IDS). Fazem análises por assinaturas ou por heurísticas comportamentais.

Assinaturas são trechos de códigos maliciosos conhecidos e as heurísticas são baseadas em comportamento diferente do esperado, indicando uma possível ameaça. Podem ser de rede (NIDS), ou de host (HIDS). Em um ambiente de riscos mais altos, podem ser usados os Intrusion Prevention Systems (IPS), que disparam ações de bloqueio proativamente.

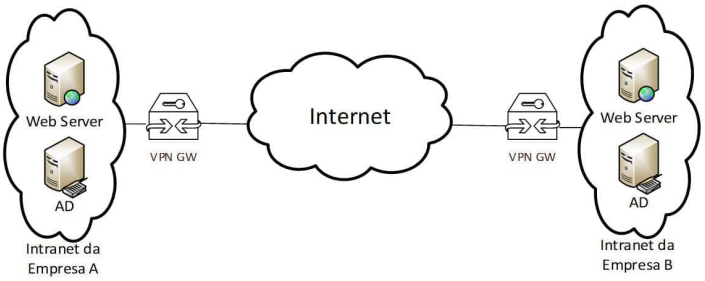
Saiba mais

Isso torna os IPS muito controversos, pois um de seus principais problemas é a possibilidade de falsos positivos, quando algo normal é encarado como um ataque, bloqueando a atividade legítima.

> **IDS dependem de treinamento, identificando ao longo do tempo o comportamento normal da rede ou do host.**

Virtual Private Network (VPN)

As VPNs se tornaram populares com a adoção do TCP/IP e a arquitetura WEB pelas empresas. Portais passaram a ser desenvolvidos para acesso remoto de seus funcionários, usando a internet como meio de acesso, o chamado VPN de acesso remoto à intranet (criando o conceito de extranet). Essa solução foi denominada **VPN site-to-site**.



VPN site-to-site.

Na imagem, vemos a interligação de duas empresas por meio de VPN gateways, que “tunelam” todo o tráfego entre ambas. Os endereços de origem e destino dos pacotes também é criptografado, garantindo a anonimidade do tráfego.

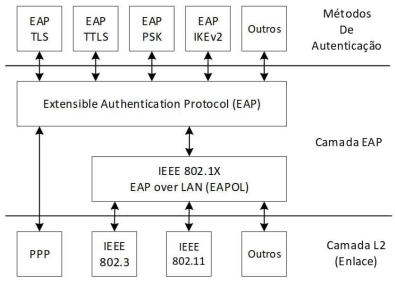
Atenção

Há vários protocolos padronizados para a operação de uma VPN, como o L2F, PPTP e L2TP, mas o IPSEC é o mais usado e será explicado adiante.

Aplicações de AAA

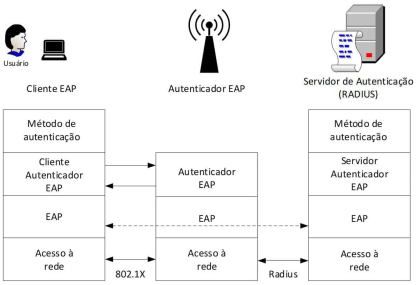
Uma das mais importantes ações da área de segurança da informação é o **controle do acesso**. Excesso de direitos é uma prática comum até mesmo em grandes empresas, em que não existe uma definição detalhada de papéis e responsabilidades que permita uma definição de direitos de acesso realista.

O padrão IEEE 802.1X é uma solução AAA que impõe uma **autorização** explícita para acesso à rede, com base em políticas previamente definidas, após a **autenticação** do usuário, feita por meio do Extensible Authentication Protocol (EAP). Além disso, mantém registro das atividades do usuário (**accounting**).



802.1X.

Como podemos ver na imagem, o EAP é um **protocolo de negociação** e **controle da autenticação**. É versátil quanto aos algoritmos usados e ao enlace, suportando redes cabeadas e sem fio.



Operação do 802.1X.

Essa imagem sintetiza as trocas em um mecanismo AAA baseado em 802.1X. O cliente EAP, desejando acessar a rede, solicita seu ingresso ao autenticador EAP. O autenticador pede as suas credenciais. A partir daí, é usado o EAPOL, um mecanismo de encapsulamento das mensagens EAP entre o cliente e o autenticador, que são repassadas para a negociação do autenticador EAP, com o servidor de autenticação Radius, em benefício do cliente EAP, de acordo com o método de autenticação configurado.

Atenção

Os mecanismos de AAA são muito importantes para a segurança de uma rede. Além do controle de acesso com mecanismos robustos de autenticação, seu registro de accounting é vital não apenas para

planejamento e gerência de recursos de rede, mas também para identificar tentativas de ataques.

Os efeitos da criptografia e das ferramentas de segurança na prática

No vídeo a seguir, abordamos a importância das ferramentas de segurança e criptografia.

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.

Falta pouco para atingir seus objetivos.

Vamos praticar alguns conceitos?

Questão 1

Suponha que, no desenvolvimento de soluções de segurança para uma rede, seja necessário verificar a presença de código malicioso em mensagens de correio eletrônico na borda da rede, de forma a evitar sua entrega ao usuário final. A solução mais indicada é

A antivírus.

B VPN.

C gateway de circuitos.

D gateway de aplicação.

E filtro de pacotes.

Parabéns! A alternativa D está correta.

Como será necessário **inspecionar** o conteúdo da mensagem, o dispositivo deve ser capaz de remontar fragmentos de pacotes e segmentos de mensagem para avaliar o conteúdo de cada mensagem. Entre as opções, a única que realiza essa tarefa é o gateway de aplicação (Application Relay).

Questão 2

Dispositivos Intrusion Prevention Systems (IPS) possuem duas características relevantes, que são

A — ação de bloqueio proativa e análise de tráfego por assinaturas e por heurísticas.

B — ação de alerta de ataques e análise de tráfego por inteligência artificial.

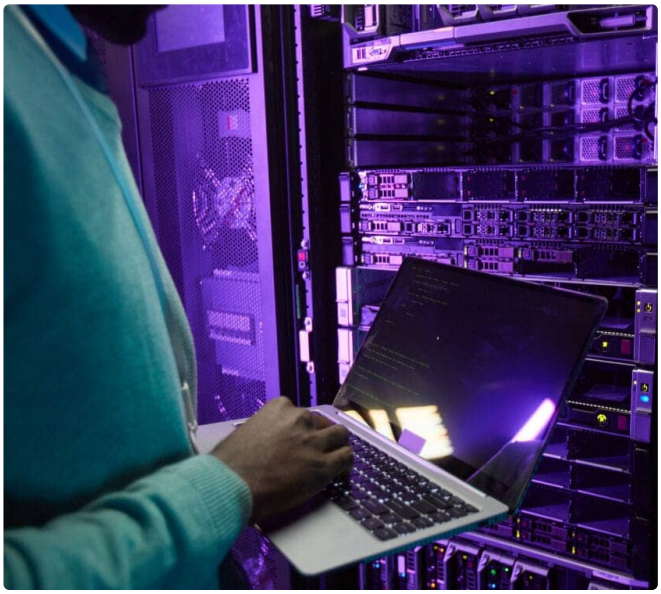
C — operação na camada de rede e análise de tráfego por regras.

D — operação na camada de transporte e controle do estado de conexões.

E — operação na camada de aplicação e desenvolvimento para aplicações específicas.

Parabéns! A alternativa A está correta.

A ação proativa é sujeita a falsos positivos, mas é uma das características do IPS, que dependem de um bom treinamento de suas heurísticas para reduzir o número de bloqueios indevidos. A opção de ação de alerta de ataques e análise de tráfego por inteligência artificial não é representativa de nenhuma solução específica; a opção de operação na camada de rede e análise de tráfego por regras se refere aos filtros de pacotes; a operação na camada de transporte e controle do estado de conexões se refere aos gateways de circuitos; e a opção de operação na camada de aplicação e desenvolvimento para aplicações específicas se refere aos relays de aplicação.



2 - Protocolos de comunicação IP, TCP, UDP, DNS e HTTP

Ao final deste módulo, você será capaz de identificar vulnerabilidades e recursos de segurança no contexto dos protocolos de comunicação IP, TCP, UDP, DNS e HTTP.

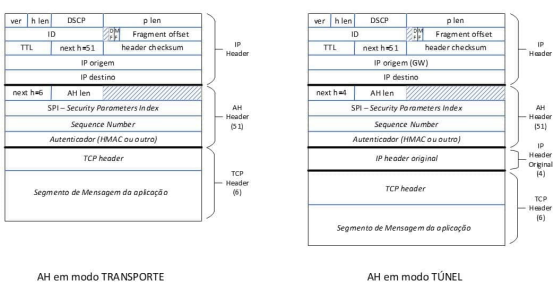
Protocolo IP – Internet Protocol

O IP tem como missão fazer que os pacotes cheguem ao seu destino de acordo com políticas de “melhor esforço”, ou seja, usando os melhores caminhos disponíveis, pacote a pacote (GOODRICH, 2013).

Atenção

O cabeçalho IP apenas possui opções para isso, não suportando criptografia ou autenticação. Regras de filtragem de pacotes precisam ser implementadas na borda da rede para evitar que pacotes, cuja origem pareça ser da rede interna, sejam descartados.

A proteção do tráfego na camada de rede pode ser feita com o IPSEC. Esse protocolo é opcional para o IPv4 e compõe o IPv6. É uma suíte com dois cabeçalhos especiais, um para **autenticação** – Authentication Header (AH) –, e o outro para a **criptografia** dos pacotes, também dando suporte à autenticação dos pacotes – Encapsulating Security Payload (ESP).



Cabeçalho AH.

A imagem ilustra os dois modos de operação do AH. No modo **transporte**, o cabeçalho IP original, o AH, o cabeçalho TCP e os dados da aplicação são autenticados. No modo **túnel**, um novo cabeçalho IP é usado, com o endereço IP de um gateway como origem dos pacotes.

No processo de autenticação, é recomendado o uso de um algoritmo como o HMAC (KENT, 2005b), com uma chave secreta compartilhada entre os pares, permitindo a verificação da integridade e da autenticidade do pacote. Fique atento a algumas observações importantes:

Modo de operação

Transporte ou túnel: é identificado no campo next h do AH. Se for modo transporte, o próximo cabeçalho já será o TCP, mas se for modo túnel, esse campo terá a identificação 4, representando um encapsulamento “IP in IP”.

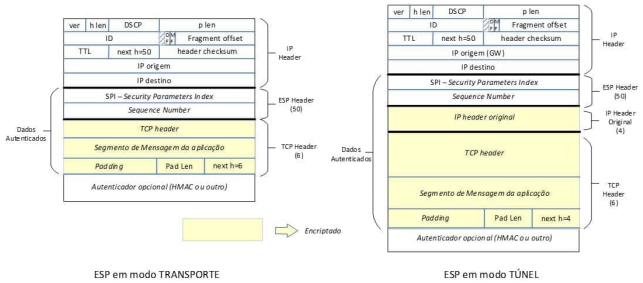
Security Parameters Index (SPI)

É um identificador de uma “Associação de Segurança (AS)” entre os pares da comunicação, que abrange todas as informações necessárias para a autenticação do pacote na origem e a verificação de integridade e autenticidade – no caso do AH – no destino. Algoritmos selecionados e chaves fazem parte dessa AS.

Sequence Number

É uma importante ação de segurança, por evitar ataques por replay. Nesses ataques, a ameaça captura um pacote legítimo para uso futuro. O uso de números de sequência autenticados evita esse tipo de ataque, uma vez que a numeração ordenada dos pacotes será alterada com a introdução de um pacote capturado anteriormente.

O cabeçalho ESP é usado quando se deseja a confidencialidade. A imagem a seguir ilustra o ESP.



Cabeçalho ESP.

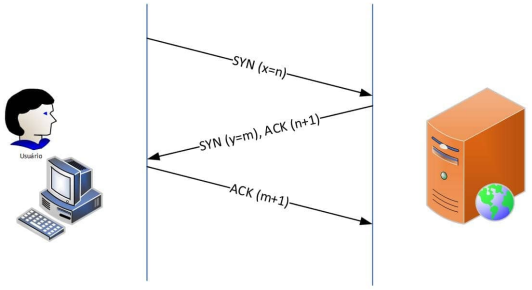
Dica

No caso do ESP, a operação de criptografia do tráfego é obrigatória e a de autenticação é opcional. Isso significa dizer que caso sejam necessárias a confidencialidade e a autenticidade/integridade basta usar o cabeçalho ESP.

Uma diferença marcante entre o AH e o ESP é que, neste último, é feito um preenchimento do segmento TCP encapsulado para múltiplos de 32 bits (4 bytes) seguidos dos campos Pad Lenght – Tamanho do padding (preenchimento) – e next h (Next header). Essa ação de preenchimento da mensagem original visa dificultar a ação de criptoanalistas, tornando a informação do tamanho real do criptograma desconhecida de uma ameaça.

Protocolo TCP - Transmission Control Protocol

O TCP tem a importante missão de garantir **confiabilidade** na comunicação entre dois hosts em uma rede IP, logo, **não confiável**. Isso tem um custo, porque é necessário que cada host defina numeradores de segmentos, buffers para transmissão e recebimento de mensagens. Não há, no TCP, qualquer implementação de segurança. Pontos de conexão (sockets) são uma combinação de endereço IP + porta, e suas **reais** identidades não são verificadas. Uma conexão TCP é estabelecida de acordo com as trocas ilustradas a seguir.



Handshake do TCP.

> **O TCP atribui numerações sequenciais de seus segmentos, a partir de um Initial Sequence Number (ISN), que é um número de 32 bits escolhido randomicamente.**

A partir daí, as numerações dos próximos segmentos serão a numeração do anterior acrescida da quantidade de bytes que ele possui, mais 1. Desse modo, é possível colocar segmentos em ordem e identificar perdas. Na imagem, o usuário pede uma conexão e identifica o seu ISN x com o valor n, porém, a mensagem de pedido de conexão está vazia, a próxima mensagem esperada é a n+1, o que ele indica com o envio da mensagem ACK=n+1. O significado literal dessa mensagem é "estou esperando o segmento n+1".

Não há autenticação dos usuários envolvidos.

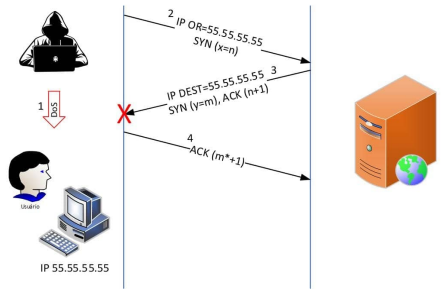
A seguir, alguns ataques típicos ao TCP serão apresentados.

TCP Hijacking

Sequestro de sessão TCP: nesse tipo de ataque uma ameaça "entra" no meio de uma comunicação TCP com intenções maliciosas. Técnicas conhecidas (GOODRICH, 2013) são a previsão de sequência TCP e o sequestro de sessão:

Previsão de Sequência TCP

Contadores de ordem dos segmentos podem ser muito simples e previsíveis (ZENG; YIN; CHEN, 2008). Esse ataque ocorre como ilustrado na imagem a seguir:



Ataque com previsão de sequência.

Inicialmente, a ameaça faz um ataque de negação em um host que ele deseja personificar (1). Envia um pedido de conexão, usando o endereço IP do usuário personificado (2). O servidor responde normalmente ao IP personificado (3), por estar na sua lista de hosts permitidos, mas a ameaça não o recebe, nem o verdadeiro possuidor do IP 55.55.55.55, por estar sob ataque de negação. Técnicas de previsão do próximo número esperado são

usadas para que o atacante “adivinhe” o número m* e envie a mensagem 4, estabelecendo a conexão.

Sequestro de sessão

Caso a ameaça e o servidor e/ou o cliente-alvo estejam em um mesmo segmento de rede, o sequestro passa a ser trivial, já que os ISNs podem ser interceptados e usados pela ameaça.

Para evitar o TCP Hijacking, são recomendadas a adoção de mecanismos obrigatórios de autenticidade e encriptação no nível de rede, com o IPSEC, ou mesmo da própria aplicação, com o SSL/TLS ou incorporados nela.

Negação de serviço TCP

A cada pedido de conexão TCP, um conjunto de recursos é alocado e registrado em uma estrutura de dados chamada TCB (TCP Control Block). Quando não há recursos suficientes, a conexão é rejeitada. Da mesma forma, caso a CPU do servidor ou roteador atacado fique sobrecarregada e mais pacotes continuem chegando, as conexões não podem sequer ser avaliadas e os pacotes são descartados por estouro de buffer. A seguir, algumas técnicas de negação são descritas.

Ataque por inundação de SYN

Ao receber uma mensagem com o flag SYN, o host monta o TCB da conexão, responde com um SYN-ACK e aguarda o ACK do solicitante, com os recursos alocados para esse fim. Se o atacante enviar muitas solicitações de conexão, o servidor rapidamente ficará sobrecarregado. Uma proposta (GOODRICH, 2013) recomenda que os segmentos sejam autenticados e a alocação de recursos seja feita após o recebimento do ACK, mas essas vulnerabilidades continuam sem solução.

Ataque otimista ACK TCP

Um atacante acompanha as conexões TCP de um host e envia ACKs para janelas ainda não completamente recebidas no destino real, fazendo com que o host aumente sua janela e receba cada

vez mais segmentos, saturando sua interface de rede e impedindo que o servidor atenda solicitações legítimas. Esse ataque é tipicamente contornado pelos IDS atuais, pela parametrização de limites (thresholds) para conexões.

Protocolo UDP – User Datagram Protocol

O protocolo UDP é muito simples. É destinado a poucos serviços de requisições e respostas curtas, em que não seja necessária a segmentação da mensagem da aplicação.

Apesar de não alocar recursos como o TCP, também pode ser usado para ataques de negação. Ao receber uma mensagem UDP para portas sem serviços em execução, o host vai responder com uma mensagem de erro, consumindo recursos. Para intensificar o efeito, usa-se o ataque reflexivo, com o uso dos **fatores de amplificação**. Se um atacante usa o IP de um alvo como origem, ele receberá as respostas do servidor UDP.

> **Para aumentar o risco, alguns comandos em aplicações que rodam sobre UDP provocam respostas grandes, ocasionando o que se chama de efeito amplificador.**

Um único pacote pode provocar respostas que consomem de 10 a 100 vezes a banda da solicitação. Alguns exemplos são o Domain Name System (DNS), serviço essencial para o funcionamento da internet, em que requisições de informações sobre uma zona pode amplificar o ataque de 28 a 54 vezes, e o Network Time Protocol (NTP), que pode amplificar em até 556,9 vezes (CISA, 2019).

A detecção e o combate a esses tipos de ataques não são suportados pelos próprios serviços ou pelo UDP. São necessários IDS para:

- Identificar RESPOSTAS UDP particularmente grandes e frequentes para determinado IP;
- Adotar o controle de estado das comunicações UDP para evitar o processamento de respostas sem a devida solicitação;
- Adotar, para os provedores de serviços baseados em UDP, políticas restritivas, identificadas com as características desses ataques, como DNS Response Rate Limit e Traffic Shapping, Ingress Filtering, para evitar o spoofing de endereços IP e manter as

aplicações servidoras atualizadas e com as funcionalidades contra esses ataques ativadas.

Domain Name System (DNS)

O DNS oferece uma estrutura distribuída globalmente, capaz de permitir resoluções de domínios em IP de forma simples e rápida. Caso o cache e o servidor local não possuam a resolução, um host poderá obtê-la recursivamente ou interativamente. Para maior agilidade e redução de tráfego, clientes e servidores armazenam as resoluções obtidas em cache, até que percam a sua validade.

O alto nível de dependência que temos do DNS é justificativa para que a sua segurança seja priorizada. Agora vamos conhecer um importante ataque ao DNS (pharming e phishing) e em seguida a principal solução em discussão: o DNSSEC.

Pharming e Phishing

Pharming é uma variação do phishing. Phishing é uma técnica de engenharia social amplamente utilizada na internet para fazer usuários fornecerem informações privadas. O pharming tem esse objetivo, com a criação de um site clonado para capturar informações sensíveis (GOODRICH, 2013).

Imagine se o cache local do usuário indica o IP de um site clonado de uma loja ou banco. Ou se o IP de um site para atualizações de um sistema tiver sido alterado para outro que contém códigos maliciosos. Poucos usuários perceberiam imediatamente estar sob ataque, e possivelmente revelariam informações ou instalariam malware em seus ambientes.

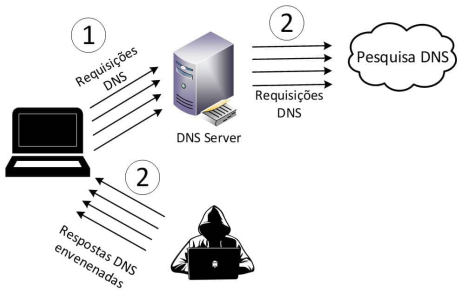
Para materializar esse ataque, uma resolução incorreta precisa acontecer, e uma das formas de se obter isso é por meio da introdução de uma resolução falsa no cache, um poisoning - envenenamento, com os seguintes passos (ANDERSON, 2017):

1. **O atacante envia múltiplas consultas para um domínio X para um servidor DNS, que as repassa para o Top Level Domain (TLD) ou domínio de nível superior.**

2. Simultaneamente, envia uma resposta para a sua própria consulta, fazendo spoofing do IP do TLD e indicando o IP de Y (malicioso) como sendo o de X.

3. O servidor DNS aceita a resposta como válida e a coloca em cache. A partir daí, todas as requisições para X indicarão o IP de Y, no qual um site clonado é disponibilizado.

Para que esse ataque possa ser bem-sucedido, a resposta do passo 2 precisa chegar antes da resposta do TLD; requisições e respostas precisam usar a mesma ID, mas a maioria das implementações disponíveis usam números sequenciais. O envenenamento também pode ser feito no cache do cliente DNS, na própria máquina do usuário. A imagem a seguir ilustra esse ataque.



Envenenamento do cache do usuário final.

Ao enviar requisições para o seu servidor, um atacante em MITM responde antes que os servidores DNS reais possam responder. Desse modo, o cache do usuário estará comprometido com uma entrada DNS envenenada. Em resumo, é possível observar que o DNS possui duas grandes vulnerabilidades em sua própria especificação, ou seja, independentemente de implementação específica (GOODRICH, 2013):

- Usar um número de 16 bits arbitrário e não protegido por mecanismos de autenticação para associar as respostas DNS às respectivas solicitações;
- A possibilidade de consultas para subdomínios inexistentes, o que deveria ser ignorado.

Comentário

Em virtude do aumento do número de ataques à servidores DNS, além de robustecimento com IDS, o DNSSEC passou a ser discutido mais intensamente.

DNSSEC

> **O DNSSEC é uma extensão do atual DNS para o uso das assinaturas digitais de todas as respostas.**

Isso implica a necessidade do uso de certificados digitais apenas pelos servidores DNS e demanda uma adoção global não apenas pelos servidores, mas principalmente pelos clientes DNS. Esse é mais um desafio nessa longa jornada de batalhas para proteção cibernética.

Protocolo HTTP – Hypertext Transfer Protocol

O protocolo HTTP foi construído para navegação web, transportando requisições e respostas que tipicamente transportam páginas em HTML. A troca de informações entre clientes e servidores é feita pelos métodos HTTP, como GET, que solicita uma página ou conteúdo, ou POST, que introduz informações em um servidor.

Tudo trafega em texto claro. O HTTP foi feito para transportar caracteres, não bits. Não havia preocupações com a confidencialidade e a autenticidade das informações. Atualmente, como o HTTP é o principal protocolo de aplicação para as aplicações web, foi desenvolvida pela Netscape a solução Secure Sockets Layer (SSL). Sem alterar o HTTP, que pode operar OVER SSL (sobre o SSL), o HTTPS. Em 1996, o IETF resolveu criar a sua solução de uso livre, o Transport Layer Security – TLS (STALLINGS, 2015).

O objetivo do SSL/TLS é oferecer os seguintes serviços para as aplicações para todo o fluxo entre um cliente e um servidor:

- Confidencialidade do tráfego – Com o uso de uma chave secreta compartilhada;
- Integridade da mensagem – Por meio de um MAC, que também faz a autenticação da mensagem;

- Fragmentação e compactação dos dados da aplicação.

O SSL/TLS não possui algoritmos criptológicos embutidos. Eles são especificados durante o handshake (negociação) entre o cliente e o servidor, de acordo com a disponibilidade de ambos e as restrições específicas para cada versão do SSL ou TLS.

O HTTPS é uma combinação entre a aplicação HTTP e o SSL/TLS, suportada pelos browsers e pelos servidores web. Além do identificador diferente – “https://” no lugar de “http://” –, a porta usada no https passa a ser a 443. Em uma sessão https, as seguintes informações estarão protegidas por encriptação e autenticação/integridade:

URL do documento

O conteúdo dos documentos e formulários (preenchidos ou não)

Cookies enviados e recebidos

Cabeçalho http

O fato de a arquitetura da internet ter sido criada em um cenário sem grandes preocupações com segurança traz grandes riscos para um ambiente sem mecanismos de controle. Em cada camada de protocolos temos vulnerabilidades documentadas, mas também temos soluções viáveis. Um bom planejamento é vital, iniciando pela evidenciação das vulnerabilidades do ambiente e pela mensuração dos impactos, de forma que possam ser escolhidas, configuradas e implementadas as melhores soluções para cada caso.

Proteção na navegação web

No vídeo a seguir, demonstramos a facilidade de se capturar informações na navegação web, e comparamos com capturas após a incorporação da segurança.

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.

Falta pouco para atingir seus objetivos.

Vamos praticar alguns conceitos?

Questão 1

Um analista de segurança resolve implementar uma VPN site-to-site com o uso do IPSEC. Para garantir a criptografia e a autenticação de todo o tráfego com o menor overhead possível, ele vai precisar usar

A o cabeçalho AH (Authentication Header).

B o cabeçalho ESP (Encapsulating Security Payload).

C os cabeçalhos AH e ESP.

D o algoritmo AES, apenas.

E o algoritmo HMAC, apenas.

Parabéns! A alternativa B está correta.

O cabeçalho AH suporta apenas autenticação. O ESP suporta a criptografia e opcionalmente a autenticação, logo, é a resposta correta. Apesar de possível, não é necessário o uso de ambos (AH e ESP), e há a ressalva de menor overhead. O AES é um algoritmo de criptografia e o HMAC é utilizado apenas para verificação de integridade e autenticidade.

Questão 2

O HTTPS é um dos protocolos mais usados atualmente na internet e tem por objetivo aumentar o nível de segurança das aplicações web. Acerca desse assunto, analise as alternativas a seguir.

I. O HTTPS não é uma versão do HTTP, e sim o próprio HTTP operando sobre o SSL ou o TLS.

II. O HTTPS permite a proteção ou não de mensagens seletivamente dentro de uma mesma sessão.

III. O HTTPS na internet demanda o uso de certificado digital do servidor web.

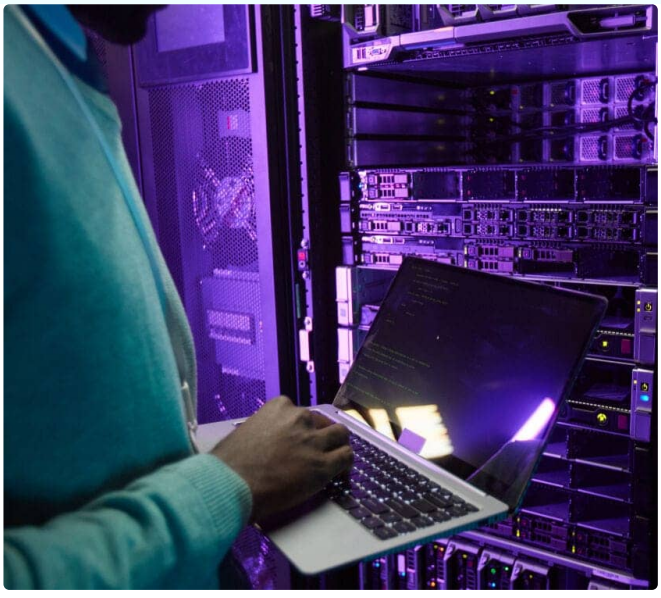
IV. As portas TCP usadas pelo HTTP e pelo HTTPS são diferentes.

Está correto o que se apresenta apenas em:

A	II.
B	I e III.
C	II e III.
D	I, II e IV.
E	I, III e IV.

Parabéns! A alternativa E está correta.

O HTTPS, que é executado na porta 443 por padrão, não é uma implementação de uma nova versão do protocolo, mas sim o HTTP, que é executado na porta 80 por padrão, empregando mecanismos de segurança, mais especificamente o SSL ou TSL, garantindo confidencialidade, integridade e autenticidade. A autenticidade é garantida por meio do emprego de certificados digitais no servidor. Além disso, todo o tráfego será submetido ao mesmo modelo segurança combinado durante o handshake. Não há possibilidade de proteção seletiva.



3 - Hardening de segurança em ambientes Linux e Windows

Ao final deste módulo, você será capaz de selecionar recursos para hardening de segurança em ambientes Linux e Windows.

Segurança em sistemas operacionais

Sistemas operacionais são ambientes com milhões de linhas de código. Infelizmente, são comuns episódios de descoberta de vulnerabilidades após meses ou anos de uma versão ser lançada. Neste módulo, vamos descrever as ações de segurança recomendáveis para os dois mais usados: Microsoft Windows e Distribuições Linux.

Segurança em ambiente windows

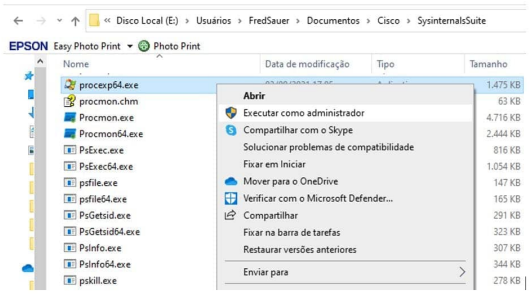
De um modo geral, a Microsoft recomenda as seguintes ações de segurança (CISCO, 2020):

- Proteções contra malware: por default, a Microsoft recomenda o Windows Defender;
- Serviços desconhecidos ou sem interface de gerência: certificar-se de que todos os serviços em execução são identificáveis e seguros;
- Encriptação de informações sensíveis;
- Adoção de políticas de segurança: regras baseadas em melhores práticas, algumas passíveis de implementação pelas ferramentas administrativas do próprio Windows;
- Firewall: usando o firewall nativo ou de terceiros, é importante que seja periodicamente revisado, para que as suas regras espelhem o desejado;
- Apuro na configuração de permissões e compartilhamentos: atribuição a cada usuário ou grupo criado apenas dos direitos mínimos necessários para cumprimento de seu papel e responsabilidades;
- Uso de autenticadores robustos: em caso de uso de senhas, que sejam fortes;
- Uso de logins de usuário com direitos especiais, em vez do administrador, limitando o alcance desse usuário e permitindo a responsabilização do agente responsável por transações realizadas.

Arquivo de registro

O Windows armazena suas informações sobre hardware, aplicações, configurações do sistema e usuários na base de dados chamada de **arquivo de registro**. É organizado em chaves e subchaves com valores armazenados que determinam como os componentes se relacionam. É editável por usuários com direitos de administrador, representando uma grande vulnerabilidade. Aplicações maliciosas, como um keylogger, podem facilmente adicionar chaves no registro, determinando seu carregamento ao boot. Uma ação comum em procedimentos de perícia forense é estudar o arquivo de registro para verificar quais aplicações são inicializadas junto com o boot do sistema, e a partir de qual diretório, visando evidenciar a ocorrência de um ataque.

Há facilidades para evitar que qualquer usuário use logins genéricos de administrador. Para iniciar qualquer aplicação como administrador, basta iniciar clicando com o botão esquerdo do mouse e selecionar “executar como administrador”, conforme ilustra a imagem a seguir.



Execução de uma aplicação como ADMINISTRADOR.

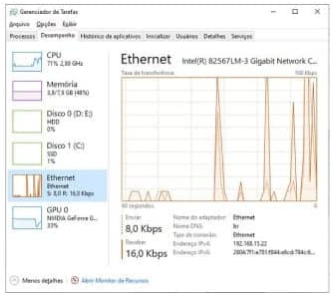
Dica

As contas de administrador e guest são desabilitadas por default, e não é recomendável a sua habilitação. A melhor prática de segurança é definir papéis e responsabilidades que demandem acesso privilegiado e conceder os direitos necessários ao colaborador responsável.

> **Para situações de crise, qualquer usuário com a credencial de administrador poderá fazer a tarefa sob supervisão.**

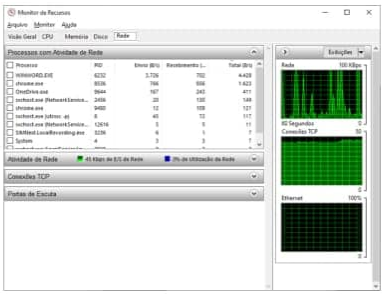
O Windows oferece a funcionalidade de grupos, que, ao serem criados, podem receber privilégios diferenciados. Ao inserir determinado usuário em um grupo, ele passa a ter os direitos desse grupo também. Esse conceito é estendido para os domínios do Windows em que cada usuário é autenticado e ganha acesso à rede e seus recursos.

Sempre que existe a suspeição de ataque e presença de malware, buscamos ferramentas para inspeção. O Windows possui duas ferramentas muito úteis para esse fim, por permitirem uma avaliação rápida das razões por um desempenho diferente do normal. Vamos conhecê-las a seguir:



Task manager (gerenciador de tarefas).

A primeira é o task manager – gerenciador de tarefas, que apresenta informações sobre processos em execução, inclusive um histórico de consumo de recursos, entre outras informações importantes.



Monitor de recursos.

A outra ferramenta é o monitor de recursos, mais detalhado nesse aspecto

O consumo de CPU, rede e memória pode indicar atividade maliciosa. É importante saber identificar **qual processo está causando um eventual aumento no consumo de recursos**, para um diagnóstico de sua malignidade.

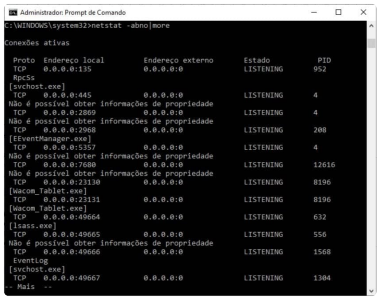
Em servidores Windows, a segurança é ainda mais desafiadora, já que o sistema operacional incorpora aplicações tipicamente usadas em uma rede, como DNS, WEB e DHCP Server, além de serviços de compartilhamento como SMB e NFS, entre outras funcionalidades que precisam ser adequadamente configuradas e monitoradas para evitar incidentes.

Aplicações importantes para o gerenciamento da segurança em ambiente Windows

A seguir, veremos aplicações importantes para o gerenciamento da segurança:

Netstat

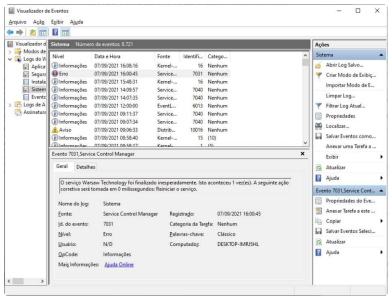
Qualquer aplicação em execução que estabelecer conexões com outras (cliente/servidor ou P2P) dará início a associações entre sockets de origem e destino visíveis por meio dessa (e de outras) ferramentas. O uso da opção <-abno> faz com que as conexões sejam diretamente associadas aos processos correspondentes, o que pode permitir a identificação de código malicioso em execução. A imagem a seguir mostra a saída desse comando.



Saída do Netstat no Windows.

Windows event viewer

Importante ferramenta que registra ocorrências referentes a aplicações, seguranças, instalações/configurações, sistemas e outros eventos. A imagem a seguir ilustra uma entrada referente ao serviço Warsaw, usado por bancos para monitorar transações feitas por computadores. Paralisações desse serviço com frequência devem ser analisadas para identificar uma eventual ação maliciosa.

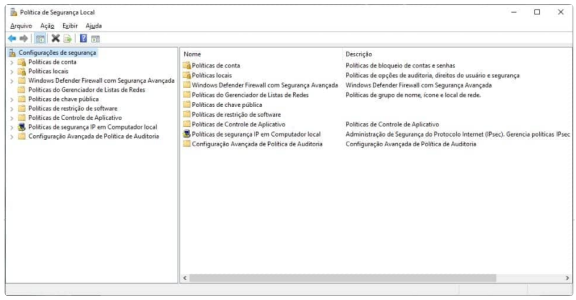


Event Viewer (Visualizador de Eventos).

Política de segurança local

Em uma rede Windows, o Active Directory (AD) é configurado por meio de domínios. Os computadores entram nesses domínios e o administrador pode configurar políticas para todos que nele ingressem. Isso também funciona em ambientes sem AD. Políticas de senha, bloqueio de tela por inatividade e várias outras funções, como gerenciamento avançado do firewall do

Windows e restrições no uso de aplicações. A imagem a seguir ilustra a ferramenta.



Políticas de segurança local.

Windows Defender

Importante e gratuita ferramenta, vem evoluindo com as novas versões de Windows. Atualmente, oferece suporte à proteção antivírus, adware, phishing, spyware e fontes confiáveis ou suspeitas.

Windows Firewall

Também vem em franca evolução. Implementa o nível mais básico de firewall, por meio de filtros de pacotes. Em configuração avançada, é possível configurar comunicação remota usando o IPSEC.

Como em qualquer ambiente operacional, existem recursos de segurança que precisam ser configurados com base em políticas descritas de acordo com as estratégias do negócio. Boas práticas também são importantes e não devem ser negligenciadas no uso de todos os recursos possíveis contra as ameaças cibernéticas.

Segurança em ambiente linux

O Linux é um sistema operacional open source. Isso significa que os códigos-fonte de tudo o que é executado é acessível e pode, inclusive, ser modificado, recompilado e substituir módulos testados e comprovadamente seguros. O fato de ser muito rápido e pequeno, quando comparado ao Windows e a outros sistemas, tornou o Linux um SO bastante usado, principalmente para servidores (CISCO, 2020).

O Linux também é uma excelente opção para a operação em Security Operation Center (SOC), devido à quantidade de aplicações voltadas para análises de segurança, como sniffers (wireshark e vários outros),

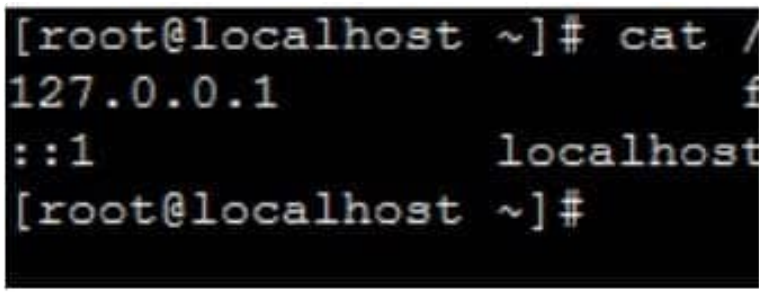
detectores de intrusos e vulnerabilidades (Nessus e Snort), entre outras opções menos populares.

> **Uma das mais marcantes características do Linux é ter quase tudo organizado e gerenciado em arquivos.**

Exemplo

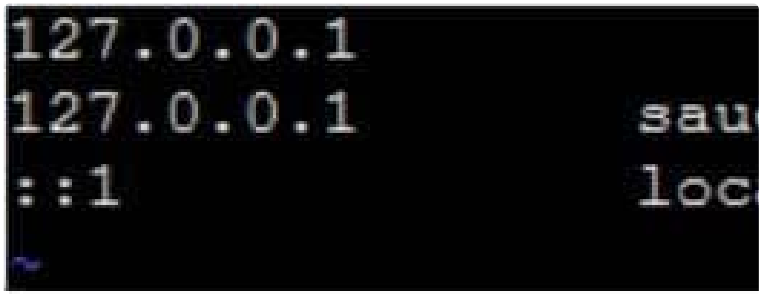
Apresentar uma informação em uma tela, para o Linux, é escrever no arquivo que representa o device correspondente (como um monitor de vídeo). Isso representa um perigo, pois qualquer um que tenha acesso a esses arquivos e às devidas permissões pode alterar suas configurações e mudar completamente o comportamento de uma aplicação ou serviço.

A imagem a seguir mostra o conteúdo do arquivo hosts, que é lido antes de qualquer consulta ao DNS. Se uma linha for acrescentada a ele, qualquer tentativa de resolução de URL para IP verificará suas entradas antes de consultar o serviço DNS.



Saída do comando CAT no arquivo /etc/hosts.

Esse arquivo apenas pode ser editado pelo root. Caso uma ameaça consiga fazê-lo, poderia fazer o que aparece na imagem a seguir.



Saída do comando cat /etc/hosts após inserção maliciosa.

A inserção de uma linha fazendo o domínio sauer.com.br apontar para 127.0.0.1 (loopback) impediria o usuário de acessar esta URL, em um

ataque de negação.

Arquivos de configuração dos serviços de um ambiente Linux ficam em formato de texto, como na imagem a seguir, referente ao serviço de definição de horário global. Números de portas usadas, permissões e restrições de uso e várias outras. Isso significa que o controle no acesso e na autorização é de suma importância para a segurança de um ambiente Linux.

```
[analyst@secOps ~]$ cat /etc/ntp.conf
# Please consider joining the pool:
#
#     http://www.pool.ntp.org/join.html
#
# For additional information see:
# - https://wiki.archlinux.org/index.php/Network_Time_Protocol_daemon
# - http://support.ntp.org/bin/view/Support/GettingStarted
# - the ntp.conf man page

# Associate to Arch's NTP pool
server 0.arch.pool.ntp.org
server 1.arch.pool.ntp.org
server 2.arch.pool.ntp.org
server 3.arch.pool.ntp.org

# By default, the server allows:
```

Arquivo de configurações do serviço NTP.

Melhores práticas para o hardening de um ambiente Linux

Melhores práticas é um termo que se refere ao uso de estratégias comprovadamente eficazes em situações de risco. E hardening é a adoção de boas práticas no sentido de robustecer um ativo, como um sistema computacional ou até mesmo um dispositivo passivo de conectividade. Podem ser adotadas medidas lógicas e físicas para esse fim.

Para um ambiente Linux, medidas gerais indicam o uso de um apurado gerenciamento de credenciais de acesso, ainda que sejam apenas senhas, acompanhado da definição de papéis e responsabilidades para cada usuário e permissões limitadas a essas definições.

A adoção de cuidados especiais para o acesso remoto também é ponto vital. Um importante recurso é o **System Security Services Daemon (SSSD)**, que gerencia o acesso remoto e a autenticação para single sign-on.

> **Sistemas Linux são oferecidos com vários serviços habilitados por default. Muitos deles serão desnecessários**

> **para o fim do sistema em instalação, logo, precisam ser desinstalados.**

Da mesma forma que em sistemas Windows, todos os dias são descobertas vulnerabilidades no Linux. Mantê-lo atualizado é o melhor caminho para reduzir a probabilidade de possuir uma vulnerabilidade conhecida.

Sintetizando as ações de boas práticas para o hardening Linux, temos o seguinte checklist (CISCO, 2020):

1. Atender requisitos de segurança física;
2. Minimizar pacotes instalados ao mínimo necessário e desinstalar os restantes;
3. Desabilitar serviços desnecessários que não possam ser desinstalados;
4. Usar SSH desabiptando o login de root via SSH caso haja demanda de acesso remoto;
5. Manter o sistema atuapzado, Kernel e pacotes;
6. Desabilitar autodetecção de USB;
7. Adotar a Multiple Factor Authentication (MFA). Usar senhas fortes, forçando trocas periódicas e não permitir reuso de senhas;
8. Revisar periodicamente os logs.

Além disso, o uso das ferramentas tradicionais de combate a incidentes, como firewall, IDS, Syslog, e outras de acordo com o nível de risco são de uso indispensável.

Os logs do Linux são ricos em informação importante, pois permitem não apenas o acompanhamento das principais transações de cada módulo do sistema, mas também um conhecimento profundo das características em operação normal – desempenho e nível típico operacional, como número de acessos e tipos de interação com hosts externos. A seguir, são listados alguns logs de especial interesse. Dependendo da distro, nomes podem variar.

/var/log/message

Diretório de logs de atividade do computador, de nível informacional ou não crítico.

/var/log/auth.log

Arquivo de log de atividades de autenticação e autorização no sistema.

/var/log/secure

Logins feitos como superusuário (sudo), SSH e erros registrados durante a operação do SSSD.

/var/log/boot.log

Mensagens durante o processo de startup.

/var/log/dmesg

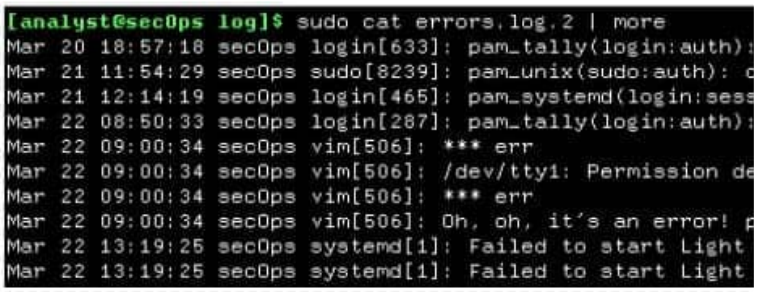
Importante diretório de logs que armazenam mensagens do kernel, sobre o hardware e seus drivers, em mais baixo nível que o administrador costuma tomar conhecimento.

/var/log/cron

Sempre que uma tarefa é programada para execução (cron job), esse diretório conterá registros referentes a cada ocorrência.

A imagem a seguir ilustra o conteúdo de um log de erros com ocorrências de tentativa de acesso por meio de contas inexistentes,

entre outras ocorrências dignas de investigação.



Extrato de um log de erros

Ambientes Windows e Linux possuem muitos recursos para controle de riscos, mas infelizmente a maioria é negligenciada e só é utilizada após a ocorrência de um incidente. Este módulo mostrou vários recursos que, com planejamento para implementação e gerenciamento durante a operação, podem reduzir os riscos e evitar grande parte dos potenciais incidentes de segurança.

Segurança no Windows e Linux

Algumas ferramentas úteis para diagnosticar a presença de ameaças são apresentadas no vídeo a seguir.

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.

Falta pouco para atingir seus objetivos.

Vamos praticar alguns conceitos?

Questão 1

O Windows possui ferramentas nativas que podem ser úteis para a verificação de um consumo anormal de CPU e memória de um computador, o que pode evidenciar um ataque em andamento. São ferramentas úteis para monitoramento de desempenho:

A Netstat e Syslog.

B IPCONFIG e Netstat.

C Task manager e Monitor de recursos.

D Visualizador de eventos e monitor de recursos.

E	Gerenciador de CPU e memória.

Parabéns! A alternativa C está correta.

Entre as ferramentas mencionadas nas questões, o Task manager e o Monitor de recursos apresentam gráficos que permitem um monitoramento em tempo real das condições de operação da CPU e da memória.

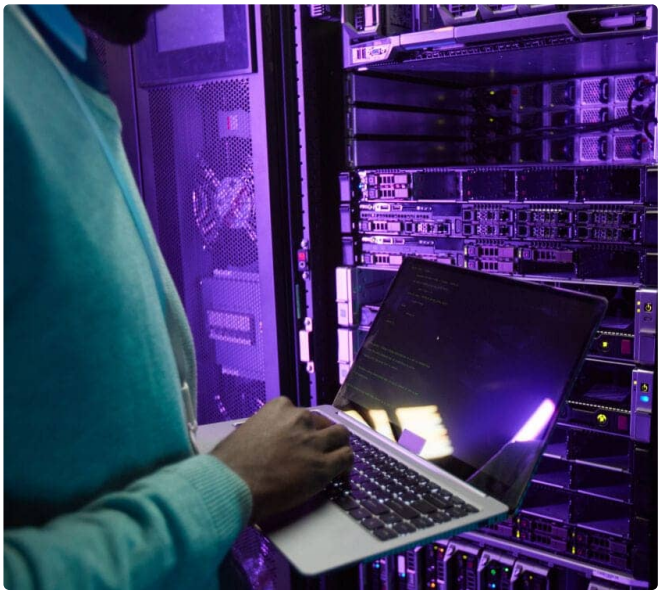
Questão 2

O Linux possui um importante recurso para o registro das ocorrências relacionadas ao sistema, às aplicações, e a várias outras atividades da operação do sistema. Esse recurso é chamado coletivamente de

A	dump.
B	arquivo de registro.
C	arquivos de configuração.
D	log.
E	messages.

Parabéns! A alternativa D está correta.

Esses arquivos de log são organizados de acordo com a natureza da ocorrência, e armazenados em um diretório próprio em uma instalação Linux (tipicamente o diretório /var/log).



4 - Estratégias de segurança para redes sem fio (wireless) e IoT

Ao final deste módulo, você será capaz de selecionar estratégias de segurança para redes sem fio (wireless) e IoT.

Segurança em redes sem fio (802.11)

Como os protocolos da internet foram concebidos para operação cabeada, a adoção de canais de rádio para tráfego de dados trouxe desafios como (GOODRICH, 2013):

Espionagem de tráfego

Bem mais simples que em ambientes cabeados, essa atividade maliciosa é de difícil detecção, porém fácil de se realizar.

Sequestro de sessão

Muito mais fácil e prático de ser realizado, pois o meio de um ataque Man in the Middle (MITM) é de livre acesso e a captura dos pacotes não demanda interceptação física. O mercado disponibiliza interfaces próprias para análises de tráfego.

Negação de serviço

É fácil de ser realizado e permite que determinada rede fique indisponível para uso.

Personificação do Access Point (AP)

Consiste em assumir a identidade de um ponto de acesso.

Intrusão

Conectar-se e usufruir da conectividade oferecida por um AP sem fio é trivial, dependendo das soluções de segurança adotadas.

Verificação de identidade

Em comparação com redes cabeadas, é bem mais complexo garantir autenticação e autorização seguras, pois o usuário não precisa estar fisicamente conectado a um ponto visível da rede, demandando, assim, ações complementares.

Atenção

É importante observar que a segurança nas soluções cabeadas está intrinsecamente ligada às camadas superiores, e toda a negociação de acesso nas redes sem fio está no próprio enlace, o que demanda ações apropriadas.

Existem quadros 802.11 com finalidades específicas de interesse para a segurança:

Quadro de autenticação

Usado para a identificação de um cliente ao AP. Também usado para a resposta do AP ao usuário.

Quadro de associação

Caso a identificação tenha sido bem-sucedida, o usuário envia esse quadro para que o AP insira essa estação no seu mecanismo de controle de acesso ao meio físico.

Quadro de beacon

Quadro de "anúncio" da existência de uma rede.

> **Uma vez autenticado e associado, quadros para desassociação, desautenticação e reassociação podem ser utilizados.**

Saiba mais

Mecanismos de segurança para redes IEEE 802.11

As redes sem fio foram impulsionadas por demandas de mercado, o que fez surgir o termo Wireless Fidelity – Wi-Fi (WI-FI ALLIANCE, 2021). Esse termo representa um "carimbo" de compatibilidade com outro dispositivo Wi-Fi, mesmo de outros fabricantes. IEEE e Wi-Fi Alliance cooperam de forma intensa na proposição de soluções de segurança descritas a seguir.

Wired Equivalent Privacy (WEP)

Esse primeiro mecanismo apresentou problemas graves e atualmente não é recomendado. O WEP disponibiliza dois modelos de associação:

Sistema aberto

O cliente usa as informações de um quadro de beacon, como o SSID, canal e taxa de operação etc. para se associar sem a necessidade de chaves.

Chave compartilhada

Nesse modelo, cliente e AP precisam ter uma chave previamente compartilhada. Ao solicitar associação, o cliente recebe do AP um "desafio" composto de uma string, que deve ser encriptada

pelo cliente com a chave compartilhada, e decriptada com sucesso pelo AP. A integridade das mensagens é provida por um CRC de 32 bits inserido antes da criptografia e a confidencialidade é atendida pelo algoritmo simétrico de fluxo RC4, e aí começam os problemas.

O RC4 usa um vetor de inicialização (IV), que é concatenado à chave compartilhada. Esse procedimento visa dar aleatoriedade à chave em cada operação criptológica. No entanto, por ainda não haver um canal seguro entre cliente e AP, este IV é enviado em claro no cabeçalho do quadro. Seu pequeno tamanho (24 bits), associado ao uso de uma chave estática possivelmente de apenas 40 bits, facilitam quebras por força bruta.

WPA

Em uma parceria importante entre a indústria e a pesquisa, o IEEE iniciou discussões no contexto da segurança em um workgroup denominado 802.11i. Enquanto isso, a Wi-Fi Alliance propunha o WPA, com ações bem objetivas para solucionar os problemas do WEP:

Problemas com o IV (initialization vector)

O IV é estendido para 48 bits e são introduzidos mecanismos para mais cuidados na escolha e verificação deles.

CRC

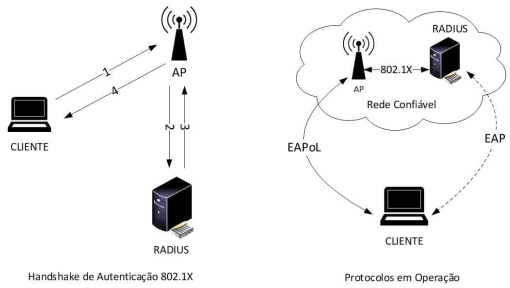
O cálculo dos CRCs é previsível. Assim, é possível alterar maliciosamente bits da mensagem e corrigir o CRC. O WPA introduz o Message Integrity Code (MIC) de 64 bits.

Chaves estáticas

O WPA não usa chaves estáticas para criptografia do tráfego, mas sim chaves temporárias geradas e distribuídas pelo AP.

Modelos de autenticação WPA

A seguir, vamos conhecer os modelos de autenticação WPA:



Handshake e operação 802.1X

- Na ilustração à esquerda, o handshake se inicia com o cliente se associando ao SSID da rede e enviando suas credenciais pessoais de login, no lugar de chaves compartilhadas.
- O AP submete as credenciais ao servidor de autenticação.
- O servidor de autenticação gera e envia uma Master Session Key (MSK), que poderá ser usada para derivar outras chaves, dependendo da configuração adotada.
- Na imagem à direita, pode-se observar que a comunicação entre o cliente e o servidor RADIUS é abstrata, porque é feita pelo AP usando o EAPoL.

Atenção

Há várias opções de autenticação. Podem ser escolhidos desde apenas um par login-senha até o uso de certificados digitais por todos os envolvidos. EAP-MD5, EAP-TLS, EAP-TTLS (EAP-Tunneled TLS) e PEAP (Protected EAP) são algumas das configurações possíveis.

WPA2

Ao finalizar as especificações do IEEE 802.11i, algumas novas adições foram feitas pelo Wi-Fi Alliance, dando origem ao WPA2. O formato final dessa solução foi denominado Robust Security Network (RSN), com as seguintes novidades (STALLINGS, 2015):

- WEP, com chaves de 40 ou 104 bits, para garantir compatibilidade com instalações e dispositivos legados;
- TKIP, Temporal Key Integrity Protocol, que suporta a confidencialidade por meio do RC4 com chaves temporárias, e a integridade com o MIC de 64 bits;
- CCMP, que criptografa os dados com o AES e o modo de cifra CTR. A integridade é suportada por meio de CMAC (CBC-MAC), inserindo uma chave secreta no autenticador. As chaves criptográficas também são temporárias, como no TKIP.

IEEE 802.11w

As soluções apresentadas até agora são voltadas para a proteção dos quadros de dados, mas não se destinam à proteção dos quadros de gerenciamento e de controle. Isso pode ser explorado trivialmente em um ataque de negação de serviço. A injeção de quadros de desautenticação de clientes legitimamente autenticados é simples porque eles não são autenticados. O 802.11w endereça essa questão (STALLINGS, 2015).

WPA3

Em 2018, o Wi-Fi Alliance anunciou o padrão WPA3, com a inserção da proteção aos quadros de gerenciamento, prevista no 802.11w, e uma modificação no processo de geração de chaves. Mantém os modos Personal e Enterprise, mas elimina o uso da PSK como chave, fazendo uso de um mecanismo denominado Simultaneous Authentication of Equals (SAE) baseado no Diffie-Hellman Exchange (DHE). Para personalizar a autenticação, a PSK e o MAC address da estação e do AP são envolvidos na geração da chave, tornando-a resistente a ataques de quebra por força bruta, com dicionário.

Segurança de dispositivos internet of things (IOT)

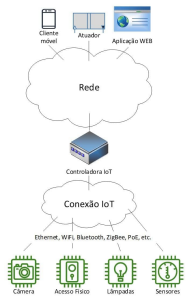
A velocidade com que novos dispositivos IoT surgem é maior do que a proposta de soluções para integrá-los às redes, sem que se transformem em um ponto de falha. Há a necessidade de um modelo de segurança que considere toda a heterogeneidade, ubiquidade, transparência e inúmeros outros desafios que a agregação desses dispositivos a uma rede impõe.

O conceito representa a possibilidade de integrar às redes dispositivos de uso doméstico e outros. O problema é que eles possuem limitações de recursos, como processador e memória e demandas de simplicidade. Ferramentas e frameworks de segurança atuais não se aplicam ao IoT (EL-GENDY e AZER, 2020).

> **São vários os problemas (OWASP, 2018) e as aplicações têm vulnerabilidades.**

Alguns dispositivos IoT não atualizam o firmware com criptografia e assinatura digital. Não há autenticação mútua robusta entre dispositivos e outros hosts. A maioria dos dispositivos permite que credenciais fracas sejam definidas.

Alarmes de segurança, sensores de pressão, temperatura, umidade e outros parâmetros físicos, câmeras de vídeo, eletrodomésticos e muitos outros têm sido integrados, com acesso aos servidores da infraestrutura.



Cenário de conectividade com IoT

Os dispositivos IoT são conectados a controladoras ou diretamente à rede por meio de soluções de LAN (como ethernet) e WLAN/WPAN, entre outras, permitindo não só a comunicação desses dispositivos com o backend (servidores de aplicações, bancos de dados, atuadores para acionamento de máquinas e outros dispositivos), como também o acesso de aplicações remotas aos próprios dispositivos.

Qualquer solução que insira ações de segurança comprometerá vários outros requisitos, como: interoperabilidade, adaptabilidade, baixo consumo de energia e ser o mais independente possível de conhecimento técnico do usuário final.

Saiba mais

Propostas atuais inserem demandas que contrariam esses requisitos, como a introdução de controladoras com competências avançadas para autenticação, autorização e accounting, certificados digitais e algoritmos criptológicos para confidencialidade do tráfego.

Em 2014, o Open Web Application Security Project (OWASP) criou o grupo de trabalho de segurança em IoT e em 2018 (OWASP, 2018) lançaram recomendações bem diretas e genéricas, ilustrados no quadro a seguir.

Nº	Vulnerabilidade	Comentários
1	Uso de senhas fracas ou autoconfiguráveis	Não devem ser aceitos dispositivos que não permitam o uso de senhas fortes configuráveis.
2	Serviços de rede inseguros disponíveis	Os dispositivos devem permitir a desabilitação de serviços desnecessários ou inseguros.
3	Interfaces inseguras no ecossistema	Dispositivos e interfaces podem possuir vulnerabilidades que sirvam como ponto de acesso aos dispositivos IoT.
4	Falta de um mecanismo de atualização	Todos os dispositivos devem possuir formas e formas de atualização rápida e segura.
5	Uso de componentes de software vulneráveis	Programas e bibliotecas usadas pelo dispositivo podem possuir vulnerabilidades.
6	Privacidade insuficiente	Dispositivos IoT tipicamente possuem capacidade para capturar todo o ambiente dos usuários, com imagens, sons e outros registros sensíveis, e os riscos precisam ser avaliados e ações de proteção criptológica adotadas.
7	Armazenamento e transferência de dados inseguros	Recursos criptológicos devem ser usados para proteger a confidencialidade do Big Data gerado pelos dispositivos IoT.
8	Dispositivos não monitorados	Dispositivos IoT devem ser integrados à arquitetura de gerenciamento da rede e monitorados atentamente.
9	Configurações padrão inseguras	Sempre planejar a configuração dos dispositivos com a segurança ao lado da funcionalidade e do desempenho.
10	Deficiência no hardening físico	Por serem portas de acesso à rede o tipicamente possuírem conveniência de uso pela simplicidade, devem ser fisicamente protegidos do acesso por pessoas não autorizadas.

Quadro: Recomendações segurança em IoT.

Extraído de: Open Web Application Security Project (OWASP).

Dica

Outra ação organizacional importante é revisar o plano de respostas a acidentes, com a introdução dos riscos com dispositivos IoT.

Redes sem fio e dispositivos IoT são de importância inquestionável e demandam recursos de segurança bem específicos. Este módulo apresentou as principais ações para reduzir os riscos e evitar incidentes de segurança nesses ambientes.

Configurando a segurança de um AP

No vídeo a seguir, abordamos as opções na configuração de segurança para um ambiente de rede sem fio.

Para assistir a um vídeo sobre o assunto, acesse a versão online deste conteúdo.

Falta pouco para atingir seus objetivos.

Vamos praticar alguns conceitos?

Questão 1

Avalie as assertivas a seguir, referentes à segurança em redes sem fio baseadas no padrão IEEE 802.11.

I - WEP e WPA são soluções com níveis equivalentes de segurança, porém, WEP é destinado a redes domésticas e WPA a redes corporativas.

II - O WPA2 foi homologado pelo consórcio de empresas Wi-Fi Alliance inserindo definições do padrão IEEE 802.11i, cuja principal agregação foi o protocolo de criptografia AES com modo de cifra CTR.

III - A solução para uso corporativo do WPA utiliza um servidor de autenticação e o protocolo EAP, permitindo que cada usuário tenha as suas próprias credenciais para acesso à rede.

É(São) verdadeira(s) a(s) assertiva(s):

A I.

B II e III.

C I e III.

D II.

E III.

Parabéns! A alternativa B está correta.

A assertiva I é falsa porque o WEP tem várias vulnerabilidades que foram corrigidas no WPA. As assertivas II e III estão corretas.

Questão 2

O fenômeno IoT traz grande comodidade, mas impõe desafios. Avalie as assertivas a seguir e a relação proposta entre elas.

I - Soluções de segurança demandam recursos criptológicos que consomem recursos computacionais (CPU, memória) adicionais aos das aplicações tradicionais, fazendo com que a adoção de segurança em ambiente IoT represente um desafio.

PORQUE

II - Por definição, equipamentos IoT devem ser diminutos, consumir pouca energia e precisar de intervenção mínima pelos usuários.

A respeito dessas assertivas, assinale a opção correta:

A As assertivas I e II são proposições verdadeiras, e a II é uma justificativa correta para a I.

B As assertivas I e II são proposições verdadeiras, mas II não é uma justificativa correta para a I.

C A assertiva I é uma proposição verdadeira, mas II é uma proposição falsa.

D A assertiva I é uma proposição falsa, mas II é uma proposição verdadeira.

E As assertivas I e II são proposições falsas.

Parabéns! A alternativa A está correta.

O objetivo do IoT é levar a conectividade a dispositivos de uso cotidiano, com a incorporação de recursos computacionais. Um dos principais requisitos é a simplicidade, para que os usuários não precisem ser técnicos, além de não onerar o objeto, com capacidade computacional desnecessária.

Considerações finais

A evolução da integração de recursos computacionais ao cotidiano humano tem trazido grandes desafios na área da segurança. Incidentes de segurança vêm ocorrendo de forma cada vez mais agressiva, chegando a impedir que pessoas e empresas possam acessar seus dados, com o objetivo de extorquir os atingidos. Apesar de já existirem leis criminalizando atividades hacker, a dificuldade de identificação dos autores torna a resolução desse problema um grande desafio.

Para os profissionais da área, cabe adotar ações preventivas e eventualmente reativas, em caso de ocorrência de incidentes. Como as ações são pontuais, desenvolvidas especificamente para um conjunto de elementos pertencentes a cada cenário de risco, é fundamental a capacidade de escolher as soluções mais adequadas para cada cenário, e desenvolver uma estratégia.

Nesse contexto, apresentamos ferramentas de segurança e ações de hardening para mitigar ataques cibernéticos, descrevendo suas características e ilustrando com exemplos.

No podcast a seguir, justificamos a importância de se adotar ferramentas de segurança como resultado de uma análise de riscos.

Para ouvir o *áudio*, acesse a versão online deste conteúdo.

Explore +

Leia o manual oficial do CERT-BR chamado **Práticas de Segurança para Administradores de Redes Internet**, sobre todas as técnicas que discutimos neste conteúdo, para se aprofundar no assunto.

Referências

ANDERSON, R. **Security Engineering**. 2nd ed. Indianapolis, IN, USA: Wiley Publishing, 2017.

CYBERSECURITY & INFRASTRUCTURE SECURITY AGENCY. **Alert (TA14-017A)** - UDP-Based Amplification Attacks. CISA, 18 dec. 2019. Consultado na internet em 1 set. 2021.

CISCO SYSTEMS. **CCNA Cybersecurity Operations 1.12.** Consultado na internet em 3 set. 2021.

EL-GENDY, S., AZER, M. A. **Security Framework for Internet of Things (IoT)**. In: 2020 15th International Conference on Computer Engineering and Systems (ICCES), p. 1-6, 2020.

GOODRICH, M. T.; TAMASSIA, R. **Introdução à segurança de computadores**. 1. ed. Porto Alegre: Bookman, 2013.

ISACA. **CISM Review Manual 2013**. 11th. ed. Schaumburg, IL, USA: Isaca, 2013.

KENT, S. **RFC 4302**: IP Authentication Header. Marina del Rey, CA, USA: RFC Editor, dec. 2005a.

KENT, S. **RFC 4303**: IP Encapsulating Security Payload (ESP). Marina del Rey, CA, USA: RFC Editor, dec. 2005b.

OWASP. **Internet of Things (IoT) Top 10 2018**. Consultado na internet em 13 set. 2021.

SAADI, M. A.; KUMAR, B. **A Review on Elliptic Curve Cryptography**. International Journal of Future Generation Communication and Networking, v. 13, n. 3, p. 1597-1601, 2020.

STALLINGS, W. **Criptografia e segurança de redes**: princípios e práticas. 6. ed. São Paulo: Pearson, 2015.

WI-FI ALLIANCE. **Wi-Fi Protected Access Security Considerations**. Consultado na internet em 11 set. 2021.

ZENG, F.; YIN, K.; CHEN, M. **Research on TCP Initial Sequence Number Prediction Method Based on Adding-weight Chaotic Time Series**. *In*: 2008 The 9th International Conference for Young Computer Scientists, p. 1511-1515, 2008.