**Fundamentos da Criptografia**

A internet é um sistema aberto e público com muitos dados fluindo por ela. Embora todos enviemos e armazenamos informações on-line, há algumas informações que optamos por manter em sigilo. Em segurança, esse tipo de dado é conhecido como informações de identificação pessoal. Informações de identificação pessoal, ou PII, são quaisquer informações que possam ser usadas para inferir a identidade de um indivíduo. Isso pode incluir coisas como o nome de alguém, informações médicas e financeiras, fotos, e-mails ou impressões digitais.

Manter privacidade das PII on-line é difícil porque são necessários os controles de segurança corretos para fazer isso. Um dos principais controles de segurança usados para proteger as informações on-line é a criptografia. A criptografia é o processo de transformar informações em uma forma que leitores não intencionais não conseguem entender. Dados de qualquer tipo são mantidos em segredo usando um processo de duas etapas: criptografia para ocultar as informações e descriptografia para exibi-las.

Imagine enviar um e-mail para um amigo. O processo começa com a coleta de dados em sua forma original e legível, conhecida como texto sem formatação. A criptografia pega essas informações e as transforma em um formato legível, conhecido como texto cifrado.

Em seguida, usamos a descriptografia para decifrar o texto cifrado de volta ao formato de texto simples, tornando-o legível novamente. Ocultar e revelar informações privadas é uma prática que existe há muito tempo. Muito antes dos computadores! Um dos primeiros métodos criptográficos é conhecido como cifra de César. Esse método recebeu o nome de um general romano, Júlio César, que governou o império romano perto do final do primeiro século aC. Ele o usou para manter privadas as mensagens entre ele e seus generais militares.

Uma chave criptográfica é um mecanismo que decifra o texto cifrado. Em nosso exemplo, a chave diria que minha mensagem é criptografada por 3 turnos. Com essas informações, você pode desbloquear a mensagem oculta.

Toda forma de criptografia depende tanto de uma cifra quanto de uma chave para garantir a troca de informações. A cifra de César não é amplamente usada hoje em dia devido a algumas falhas importantes. Um diz respeito à cifra em si. O outro está relacionado à chave.

Essa cifra específica depende inteiramente dos caracteres do alfabeto romano para ocultar informações.

Considere uma mensagem escrita usando o alfabeto inglês, que tem apenas 26 caracteres. Mesmo sem a chave, é muito simples decifrar uma mensagem protegida com a cifra de César mudando as letras em 26 maneiras diferentes.

Na segurança da informação, essa tática é conhecida como ataque de força bruta, um processo de tentativa e erro de descoberta de informações privadas.

A outra grande falha da cifra de César é que ela depende de uma única chave. Se essa chave foi perdida ou roubada, não há nada que impeça alguém de acessar informações privadas. Manter o controle adequado das chaves criptográficas é uma parte importante da segurança. Para começar, é importante garantir que essas chaves não sejam armazenadas em locais públicos e compartilhá-las separadamente das informações que elas irão descriptografar.

A cifra de César é apenas um dos muitos algoritmos usados para proteger a privacidade das pessoas. Devido às suas limitações, contamos com algoritmos mais complexos para proteger as informações on-line.

**Infraestrutura de chave pública**

Os computadores usam muitos algoritmos de criptografia para enviar e armazenar informações on-line. Todos eles são úteis quando se trata de ocultar informações privadas, mas somente desde que suas chaves estejam protegidas. Você pode imaginar ter que acompanhar as chaves de criptografia que protegem todas as suas informações pessoais on-line? Nem eu, nem precisamos, graças a algo conhecido como infraestrutura de chave pública.

Infraestrutura de chave pública, ou PKI, é uma estrutura de criptografia que protege a troca de informações on-line. É um sistema amplo que torna o acesso às informações rápido, fácil e seguro.

A PKI é um processo de duas etapas. Tudo começa com a trocar informações criptografadas. Isso envolve criptografia assimétrica, criptografia simétrica ou ambas. A criptografia assimétrica envolve o uso de um par de chaves públicas e privadas para criptografia e descriptografia de dados. Isso poderia ser visto como uma caixa que pode ser aberta com duas chaves. Uma chave, a chave pública, só pode ser usada para acessar o slot e adicionar itens à caixa. Como a chave pública não pode ser usada para remover itens, ela pode ser copiada e compartilhada com pessoas de todo o mundo para adicionar itens. Por outro lado, a segunda chave, a chave privada, abre totalmente a caixa, para que os itens internos possam ser removidos. Somente o dono da caixa tem acesso à chave privada que a desbloquea.

O uso de uma chave pública permite que as pessoas e os servidores com os quais o usuário está se comunicando vejam e enviem informações criptografadas que somente o usuário pode descriptografar com sua chave privada. Esse sistema de duas chaves torna a criptografia assimétrica uma forma segura de trocas informações on-line; no entanto, também retarda o processamento.

A criptografia simétrica, por outro lado, é uma abordagem mais rápida e simples para o gerenciamento de chaves. A criptografia simétrica envolve o uso de uma única chave secreta para trocar informações.

Supondo que a caixa esteja trancada novamente. Em vez de duas chaves, a criptografia simétrica usa a mesma chave. O proprietário pode usá-lo para abrir a caixa, adicionar itens e fechá-la novamente. Quando quiserem compartilhar o acesso, eles podem dar a chave secreta para que qualquer outra pessoa faça o mesmo. A troca de uma única chave secreta pode tornar as comunicações na web mais rápidas, mas também as torna menos seguras.

A PKI usa a criptografia assimétrica e simétrica, às vezes em conjunto. Tudo depende se a velocidade ou a segurança são prioridade. Por exemplo, aplicativos de bate-papo móvel usam criptografia assimétrica para estabelecer uma conexão entre as pessoas no início de uma conversa quando a segurança é a prioridade. Depois, quando a velocidade das comunicações de ida e volta é a prioridade, a criptografia simétrica assume o controle.

Uma vulnerabilidade em comum que as criptografias assimétricas e simétricas compartilham se dá quando estão estabelecendo confiança entre remetente e destinatário. Ambos os processos dependem do compartilhamento de chaves que podem ser mal utilizadas, perdidas ou roubadas. Os computadores não são naturalmente equipados para fazer a distinção de quem é de confiança ou não. É aí que a segunda etapa da PKI se aplica. A PKI aborda a vulnerabilidade do compartilhamento de chaves estabelecendo confiança usando um sistema de certificados digitais entre computadores e redes.

Um certificado digital é um arquivo que verifica a identidade do detentor da chave pública. A maioria das informações on-line é trocada usando certificados digitais. Usuários, empresas e redes mantêm um e trocam quando comunicam informações on-line como forma de sinalizar confiança.

Uma empresa que esteja prestes a lançar seu site e queira obter um certificado digital. Quando eles registram seu domínio, a empresa de hospedagem envia determinadas informações para uma autoridade certificadora confiável, ou AC. As informações fornecidas geralmente são coisas básicas, como o nome da empresa e o país onde sua sede está localizada. Uma chave pública para o site também é fornecida. A autoridade certificadora então usa esses dados para verificar a identidade da empresa. Quando confirmado, o AC criptografa os dados com sua própria chave privada. Por fim, eles criam um certificado digital que contém os dados criptografados da empresa. Ele também contém a assinatura digital da CA para provar que é autêntica.

Criptografia simétrica e assimétrica

Tipos de criptografia simétrica é o uso de uma única chave secreta para trocar informações. Como ela usa uma única chave para criptografia e descriptografia, o remetente e o destinatário devem conhecer a chave secreta para bloquear ou desbloquear a cifra.

A criptografia assimétrica é o uso de um par de chaves pública e privada para criptografia e descriptografia de dados. Ela usa duas chaves separadas: uma chave pública e uma chave privada. A chave pública é usada para criptografar dados, e a chave privada os descriptografa. A chave privada os descriptografa. A chave privada é fornecida a usuários com acesso autorizado.

**A importância do comprimento da chave**

As cifras vulneráveis a ataques de força bruta, que usam um processo de tentativa e erro para descobrir informações privadas. Essa tática é o equivalente digital a tentar todos os números em uma combinação de fechadura para tentar encontrar o número certo. Na criptografia moderna, comprimentos de chave mais longos são considerados mais seguros. Comprimentos de chave mais longos significam mais possibilidades que um invasor precisa tentar para desbloquear uma cifra.

A desvantagem de ter chaves de criptografia longas é o tempo de processamento mais lento. Embora os comprimentos de chave curtos sejam geralmente menos seguros, sua comparação é muito mais rápida. Fornecer comunicação rápida de dados on-line, ao mesmo tempo, manter informações seguras é um ato de equilíbrio delicado.

**Algoritmos aprovados**

Muitos aprovados da Web usam combinação de criptografia simétrica e assimétrica. É assim que eles equilibram a experiência do usuário com a proteção de informações.

**Algoritmos simétricos**

O triple DES (3DES) é conhecido como um cifra de bloco devido à maneira como converter texto simples em texto cifrado em “blocos” suas origens remontam ao padrão de criptografia de dados (DES), que foi desenvolvido no início da década de 1970. O DES foi um dos primeiros algoritmos de criptografia simétrica que gerou chaves de 64 bits, embora apenas 56 bits sejam usados para criptografia. Um bit é a menor unidade de medida de dados em um computador. O triplo DES gera chave três vezes mais longas. O Triple DES aplica o algoritmo DES três vezes, usando três chaves diferentes de 56 bits. Isso resulta em um comprimento de chave efetivo de 168 bits. Apesar das chaves mais longas, muitas organizações estão deixando de usar o triple DES devido a limites na quantidade de dados que podem ser criptografados. No entanto, é provável que o triples DES continue sendo usado para fins de compatibilidade com versões anteriores.

Padrão de criptografia avançada (AES) é um algoritmos simétricos mais seguros da atualidade. O AES gera chaves de 128, 192 ou 256 bits. Chaves criptográficas desse tamanho são consideradas seguras contra ataques de força bruta. Estima-se que a força bruta de uma chave AES de 128 bits poderia levar bilhões de anos para um computador moderno.

**Geração de Chave**

Esses algoritmos devem ser implementados quando uma organização escolhe um deles para proteger os dados. Uma maneira de fazer isso é usar o OpenSSL, que é uma ferramenta de linha de comando de código aberto que pode ser usada para gerar chaves públicas e privadas. O OpenSSL é comumente usado por computadores para verificar certificados digitais que são trocados como parte da infraestrutura de chave pública (PKI).

Obs: O OpenSSL é apenas uma opção. Há várias outras disponíveis que podem gerar chaves para qualquer um desses algoritmos comuns.

No início de 2014, o OpenSSL divulgou uma vulnerabilidade, conhecida como bug Heartbleed, que expôs dados confidenciais na memória de sites e aplicativos. Embora versões não corrigidas do OpenSSL ainda estejam disponíveis, o bug Heartbleed foi corrigido no final daquele ano (2014). Atualmente, muitas empresas usam as versões seguras do OpenSSL para gerar chaves públicas e privadas, o que demonstra a importância de usar softwares atualizados.

Obscuridade não é segurança

No mundo da criptografia, é preciso provar que uma cifra é inquebrável antes de afirmar que ela é segura. De acordo com o princípio de Kerchoffs, a criptografia deve ser projetada de forma que todos os detalhes de um algoritmo, exceto a chave privada, possam ser conhecidos sem sacrificar sua segurança. Por exemplo, é possível acessar todos os detalhes sobre como a criptografia AES funciona on-line e, ainda assim, ela é inviolável.

Ocasionalmente, as organizações implementam seus próprios algoritmos de criptografia personalizados. Houve casos em que esses sistemas criptográficos secretos foram rapidamente decifrados depois de se tornarem públicos. Um sistema criptográfico não deve ser considerado seguro exibir sobre seu funcionamento.

**A criptografia está em toda parte**

As empresas usam criptografia simétrica e assimétrica. Elas geralmente trabalham em equipe, equilibrando a segurança com a experiência do usuário.

Por exemplo, os sites tendem a usar criptografia assimétrica para proteger pequenos blocos de dados que são importantes. Os nomes das senhas geralmente são protegidos com criptografia assimétrica durante o processamento de solicitações de login. Depois que o usuário obtém acesso, o restante da sessão na web geralmente passa a usar a criptografia simétrica por causa da velocidade.

O uso de criptografia de dados como essa é cada vez mais exigido por lei. Regulamentos como o Federal Information Processing Standards (FIPS 140-3) e o Regulamento Geral de Proteção de Dados (GPDR ) descrevem como os dados devem ser coletados, usados e tratados. Alcançar a conformidade com qualquer uma dessas regulamentações é fundamental para demonstrar aos parceiros de negócios e aos governos que os dados dos clientes são tratados com responsabilidade.

Conhecer os conceitos básicos de criptografia é importante para todos os profissionais de segurança. A criptografia simétrica depende de uma única chave secreta para proteger dados. Por outro lado, a assimétrica usa um par de chaves pública e privada. Seus algoritmos de criptografia criam tamanhos de chave diferentes. Os dois tipos de criptografia são usados para atender às normas de conformidade e proteger dados on-line.

**Não repúdio e hash**

As chaves criptográficas são vulneráveis à perda ou a roubo, o que pode colocar em risco informações confidenciais.

Uma função hash é um algoritmo que produz um código que não pode ser descriptografado. Diferentemente dos algoritmos assimétricos e simétricos, as funções são processos unidirecionais que não geram chaves de descriptografia. Em vez disso, esses algoritmos produzem um identificador exclusivo conhecido como valor de hash ou resumo.

Imagine que uma empresa tenha um aplicativo interno usado pelos funcionários e armazenado em um drive compartilhado. Depois de passar por uma função de hash, o programa recebe seu valor de hash. Por exemplo, esse valor de hash criado é relativamente curto e foi criado com a função de hash MD5. Geralmente, as funções de hash padrão que produzem hashes mais longos são preferidas por serem mais seguras. Logo em seguida, imagine que um invasor substitua o programa por uma versão modificada que executa ações maliciosas. O programa malicioso pode funcionar como o original. No entanto, se uma linha de código for diferente da original, ela produzirá um valor de hash diferente. Ao comparar os valores hash, pode-se validar se os programas são diferentes. Os atacantes costumam usar truques como esse porque são facilmente ignorados. Felizmente,os valores de hash ajudam a identificar quando algo assim está acontecendo.

Em segurança, os hashes são usados principalmente como uma forma de determinar a integridade de arquivos e aplicativos.

Integridade de dados está relacionada à precisão e consistência das informações. Isso é conhecido como não repúdio, o conceito de que a autenticidade das informações não pode ser negada.

As funções de hash são controles de segurança importantes que possibilitam a comprovada integridade dos dados. Os analistas os usam com frequência. Uma maneira de fazer isso é encontrar o valor de hash de arquivos ou aplicativos e compará-los com arquivos maliciosos conhecidos.

Pode-se usar a linha de comando para do Linux para gerar valor de hash para qualquer arquivo no computador.

Exemplo:

sha256sum newfile.txt;

**A evolução das funções Hash**

As funções hash são controles importantes que fazem parte da estratégia de segurança de todas as empresas. O hash é amplamente usado para autenticação de não repúdio, o conceito de que a autenticidade das informações não pode ser negada.

As funções hash são algoritmos que produzem um código que não pode ser descriptografado. As funções hash convertem informações em um valor exclusivo que pode ser usado para determinar sua integridade.

**Origens de Hash**

As funções do hash existem desde os primórdios do computador. Elas foram criadas originalmente como uma forma de pesquisar dados rapidamente. Desde o início, esses algoritmos foram projetados para representar dados de qualquer tamanho com valores pequenos, de tamanho fixo, ou resumos. Usando uma tabela hash, que é uma estrutura de dados usada para armazenar e referenciar valores hash, esses pequenos valores se tornaram uma forma mais segura e eficiente de os computadores referenciarem dados.

Uma das primeiras funções hash é a Message Digest 5, mais comumente chamada conhecida como MD5. O professor Ronald Rivest, do MIT, desenvolveu a MD5 no início dos anos 90 como uma forma de verificar se um arquivo enviado por uma rede correspondia ao arquivo de origem.

Quer seja usado para converter um único e-mail ou o código-fonte de um aplicativo, o MD5 funciona convertendo dados em um valor de 128 bits. Talvez você se lembre de que um bit é a menor unidade de medida de dados em um computador. Os bits podem ser 0 ou 1. Em um computador, os bits representam a entrada do usuário de uma forma que os computadores podem interpretar. Em uma tabela hash, isso aparece como uma string de 32 caracteres. Alterar qualquer coisa no arquivo de origem de um valor de hash totalmente novo.

Em geral, quanto maior o valor do hash, mais seguro ele é. Não foi muito depois da criação do MD5 que os profissionais de segurança descobriram que os resumos de 128 bits resultavam em uma grande vulnerabilidade.

Quanto maior o valor de hash, mais seguro ele é. Não foi muito depois da criação do MD5 que os profissionais de segurança descobriram que os resumos de 128 bits resultavam em uma grande vulnerabilidade.

**Colisão de hashes**

Uma das falhas do MD5 é uma característica de todas as funções do hash. Os algoritmos de hash mapeiam qualquer entrada, independentemente de seu comprimento, em um valor de tamanho fixo de letras e números.

Os valores MD5 são limitados a 32 caracteres de comprimento. Devido ao tamanho da saída, o algoritmo é considerado vulnerável à colisão de hashes, uma instância em que diferentes entradas produzem o mesmo valor de hash. Como os hashes são usados para autenticação, uma colisão de hashes é semelhante a copiar a identidade de alguém. Os atacantes podem realizar ataques de colisão para se passar por dados autênticos de forma fraudulenta.

**Hash de Última Geração**

Para evitar o risco de colisão de hashes, eram necessárias funções que gerassem valores mais longos, as deficiências do MD5 deram lugar a um novo grupo de funções conhecidas como Secure Hashing Algorithms , ou SHAs.

O NIST aprova cada um desses algoritmos. Os números ao lado de cada função de SHA indicam o tamanho de seu valor de hash em bits. Com exceção do SHA-1, que produz um resumo de 160 bits, esses algoritmos são considerados resistentes a colisões. No entanto, isso não os torna invulneráveis a outros exploits. Cinco funções compõem a família de algoritmos SHA:

SHA-1

SHA-224

SHA-256

SHA-384

SHA-512

**Armazenamento seguro de senhas**

Normalmente, as senhas são armazenadas em um banco de dados, onde são mapeadas para um nome e usuário. O servidor recebe uma solicitação que contém as credenciais fornecidas pelo usuário. Em seguida, ele procura o nome de usuário no banco de dados, compara os com a senha fornecida e verifica se há correspondência antes de conceder o acesso.

Esse é um sistema seguro, a menos que um ataque obtenha acesso ao banco de dados do usuário. Se as senhas forem armazenadas em texto simples, um atacante poderá roubar essas informações e usá-las para acessar recursos da empresa. O hash acrescenta uma camada adicional de segurança. Como os valores de hash não podem ser revertidos, um ataque não conseguiria roubar as credenciais de login de alguém se conseguisse obter acesso ao banco de dados.

**Tabela arco-íris**

Uma tabela arco-íris é um arquivo de valores hash pré-gerados e seu texto simples associado. Eles são como dicionários de senhas fracas. Os atacantes capazes de obter o banco de dados de senhas de uma organização podem usar uma tabela arco-íris para compará-las com todos os valores possíveis.

**Adicionando um pouco de sal**

As funções com resumos maiores são menos vulneráveis a ataques de colisão e de tabela arco-íris.

O sal é uma proteção adicional usada para fortalecer as funções hash. Um salt é uma sequência aleatória de caracteres que é adicionada aos dados antes do hash. Os caracteres adicionais produzem um calor de hash mais exclusivo, tornando os dados que receberam o sal resistentes a ataques de tabela arco-íris.

Por exemplo, um banco de dados que contenham senhas pode ter várias entradas com hash para a senha “password”, se essas senhas fossem todas adicionadas com o incremento, cada entrada seria completamente diferente. Isso significa que um ataque usando uma tabela arco-íris não conseguiria encontrar valores correspondentes para “password” no banco de dados.

Mais informações:

https://voitto.com.br/blog/artigo/o-que-e-hash-e-como-funciona