**A6 Exposição de Dados Sensíveis**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Agentes de Ameaça | Vetores de Ataque | Vulnerabilidades de Segurança | | Impactos Técnicos | Impactos no Negócio |
| Específico da Aplicação | Exploração DIFÍCIL | Prevalência RARA | Detecção MÉDIA | Impacto SEVERO | Específico do Negócio / Aplicação |
| Considere quem pode ter acesso aos seus dados sensíveis e *backups* desses dados. Incluindo os dados em repouso, em tráfego, e até mesmo nos *browsers* dos seus clientes. Inclua as ameaças externas bem como as ameaças internas. | Os atacantes normalmente não partem diretamente a criptografia. Eles exploram de outra forma, tal como roubar chaves, aplicar ataques do tipo *man-in-the-middle*, ou roubar dados em texto claro fora do servidor, enquanto transitam, ou a partir do *browser* do utilizador. | A falha mais comum é simplesmente não encriptar dados sensíveis. Quando a criptografia é utilizada, a geração e a gestão de chaves é fraca, além da utilização de algoritmos e técnicas de hashing fracos. Vulnerabilidades no *browser* são comuns e fáceis de detectar, mas são difíceis de explorar em larga escala. Atacantes externos tem dificuldade em detectar falhas no lado do servidor, devido ao acesso limitado e também porque são geralmente mais difíceis de explorar. | | A falha frequentemente compromete todos os dados que deveriam ter sido protegidos. Normalmente, essas informações incluem dados sensíveis tais como registros médicos, credenciais de acesso, dados pessoais, cartões de crédito, etc. | Considere o valor de negócio dos dados perdidos e o impacto para sua reputação. Qual é a sua responsabilidade legal se estes dados forem expostos? Considere também os danos à sua reputação. |

**Estou vulnerável?**

A primeira coisa que se deve determinar é quais são os dados sensíveis o suficiente para exigir proteção extra. Por exemplo, palavras-passe, números de cartão de crédito, registros médicos e informações pessoais devem ser protegidas. Para todos esses dados:

1. Qualquer um desses dados é armazenado em texto claro a longo prazo, incluindo backups de dados?

2. Qualquer um desses dados é transmitido em texto claro, internamente ou externamente? O tráfego na internet é especialmente perigoso.

3. Algum algoritmo de criptografia utilizado é fraco ou defasado?

4. As chaves criptográficas geradas são fracas, ou elas possuem um gerenciamento ou rodízio de forma adequada?

5. Algumas diretivas de segurança do *browser* ou cabeçalhos estão ausentes quando os dados sensíveis são fornecidos/enviados ao navegador? Para um conjunto mais completo de problemas a serem evitados, consulte áreas do ASVS de Criptografia (V7), Proteção de dados (V9), e SSL (V10).

**Como faço para evitar?**

Os perigos completos da criptografia insegura, o uso de SSL e proteção de dados estão muito além do escopo do Top 10. Dito isto, no mínimo, implemente as seguintes recomendações:

1. Considerando que pretende proteger os dados de ameaças (como por exemplo, ataque interno ou de um utilizador externo), certifique-se que encripta todos os dados sensíveis em repouso e em trânsito de uma forma que iniba estas ameaças.

2. Não armazene dados sensíveis desnecessariamente. Descarte-os o mais rápido possível. Dados que você não tem não podem ser roubados.

3. Certifique-se que o nível utilizado nos algoritmos e chaves são fortes, e que o gestão das chaves está aplicada adequadamente. Considere utilizar módulos criptográficos validados pela norma FIPS-140.

4. Certifique-se que as palavras-passe são armazenadas com um algoritmo projetado especialmente para a proteção de palavras-passe, como o bcrypt, PBKDF2 ou scrypt.

5. Desabilite o preenchimento automático em formulários de recolha de dados sensíveis e desabilite a cache de páginas que contenham dados sensíveis.

**Exemplos de Cenários de Ataque**

Cenário #1: Uma aplicação encripta números de cartão de crédito numa base de dados recorrendo à encriptação automática da base de dados. No entanto, isso significa que também desencripta esses dados automaticamente quando recuperados, permitindo uma falha de injeção SQL para recuperar os números de cartão de crédito em texto claro. O sistema deveria ter encriptado os números de cartão de crédito através de uma chave pública, e só permitir a desencriptação por aplicações de back-end com a chave privada.

Cenário #2: Um site simplesmente não usa SSL em todas as páginas autenticadas. O atacante simplesmente monitora o tráfego de rede (como em uma rede wireless aberta), e rouba o cookie de sessão do utilizador. O atacante então reproduz este cookie e rouba a sessão do utilizador e acede aos dados privados do mesmo.

Cenário #3: A base de dados de palavras-passe dos utilizadores utiliza hashes simples (unsalted) para armazenar as palavras-passe de todos. Uma falha de upload de ficheiros permite que um atacante recupere o ficheiro de palavras-passe. Todos os hashes simples poderão ser expostos através de uma rainbow table de hashes pré-calculados.

**Referências**

**OWASP** - Para um conjunto mais completo de requisitos, consulte

Requisitos do ASVS na Criptografia (V7), Proteção de Dados (V9) e Segurança das Comunicações (V10)

• OWASP Cryptographic Storage Cheat Sheet

• OWASP Password Storage Cheat Sheet

• OWASP Transport Layer Protection Cheat Sheet

• OWASP Testing Guide: Chapter on SSL/TLS Testing

**Externas**

• CWE Entry 310 on Cryptographic Issues

• CWE Entry 312 on Cleartext Storage of Sensitive Information

• CWE Entry 319 on Cleartext Transmission of Sensitive Information

• CWE Entry 326 on Weak Encryption