Week 8

Andre Rodrigues - up201505639
Pedro Antunes - up201507254

Consolidation questions

1 - Explain what is an authenticated cipher and what security means for this primitive.

Cifras de Autenticação são outra primitiva criptográfica que permite produzir uma tag de autenticação simultaneamente ao processo de cifrar uma mensagem, ou seja, um algoritmo combina ambas funcionalidades numa só.

Esta primitiva fornece-nos a confidencialidade e a integridade da mensagem bem como a autenticação do emissor. Contrastando com a falta de confidencialidade por parte das “keyed hashes” e com a falta de integridade e autenticação por parte das cifras convencionais.

2 - Show that the Encrypt-and-Mac construction is insecure if used with a deterministic Mac.

Como nesta construção o MAC é feito sobre o plaintext se usarmos um MAC determinístico, ao invés de uma PRF, então a tag poderá revelar informação acerca do P o que poderá tornar fácil obter P. E outro problema, é que um atacante conseguirá identificar a repetição de plaintexts enviados, depois de este coletar vários pares (plaintext, tag) porque a tag identifica cada plaintext e como é determinística então o seu output não irá variar para dois plaintexts similares.

3 - What are the advantages of Encrypt-then-Mac?

Uma das vantagens é que o receptor após computar o MAC sobre o ciphertext não precisa de decifrar o mesmo se detectar que a tag é inválida, significando que o ciphertext foi corrompido. Logo, temos integridade do ciphertext e por consequência também temos integridade do plaintext.

Outra vantagem é o facto de que com esta construção o MAC não fornece informação sobre o plaintext, porque assumindo que a cifra é computada de forma a que o seu output pareça aleatório, então o MAC também parecerá aleatório.

4 - What is an AEAD and why is this the most adequate primitive for symmetric secure channels.

AEAD significa encriptação autenticada de dados associados e permite associar informação visível para todos com informação cifrada de forma a que ambos estejam autenticados, ou seja, basta A (dados associados) ou C (ciphertext) serem corrompidos que o processo de validação não será executado com sucesso porque a tag não será a mesma e portanto a mensagem será descartada.

Esta primitiva é muito útil e é bastante usada por exemplo em pacotes onde os dados associados correspondem aos metadados e contém o IP, porta, protocolo, etc.

5 - Find out on the Internet what are the two recommended AEAD constructions in TLS 1.3 (Section 9.1 of the RFC). What do they have in common?

TLS_AES_128_GCM_SHA256

TLS_CHACHA20_POLY1305_SHA256

Guided practical assignment

1 - Use Python to encrypt a file with AES-GCM

- O programa irá gerar a chave e o nonce e irá ler um valor fornecido no input.
- Depois irá encriptar o ficheiro e irá escrever o ciphertext em et.txt e a tag em tag.txt. Para auxiliar o processo de desencriptação irão ser criados outros ficheiros.
- $ python AES_GCM.py

2 - Make sure you can decrypt it with openSSL (if the command line does not support AEAD in your machine, use this toolhttps://github.com/jforissier/aesgcm).

- O binário “aesgcm” deverá estar no diretório corrente.
- $ ./aesgcm dec -key key.txt -iv iv.txt -in et.txt -out dt.txt -tag tag.txt
- Os ficheiros txt são fornecidos pelo script python anterior.
- Conseguimos fazer a desencriptação com sucesso, originando o ficheiro dt.txt com o plaintext.

3 - Modify the encrypted file

- No ficheiro AES_GCM.py existe uma opção de alterar o ciphertext após este ter sido gerado. Iremos usar essa opção.

4 - See if you can still decrypt it with openSSL

- $ ./aesgcm dec -key key.txt -iv iv.txt -in et.txt -out dt.txt -tag tag.txt
- aesgcm: Decrypt failed

5 - How would this be different if you were using AES-CTR?

As cifras não garantem a integridade da informação, pois não têm como saber se um ciphertext foi alterado ou não, portanto o AES-CTR iria tentar decifrar a cifra, obtendo valores que não estavam no plaintext original ou então obtendo erros. Com o AES-GCM isso não acontece porque o algoritmo verifica que para um mesmo nonce, key e ciphertext usado na fase de cifrar, se calcularmos o valor da tag ela tem de ser igual ao valor calculado na fase de cifrar daí o algoritmo concluir que o ciphertext foi alterado e portanto aborta a fase de decifrar.