Week 11

Andre Rodrigues - up201505639
Pedro Antunes - up201507254

Consolidation questions

1 - Why do most practical uses of cryptography separate clearly long-term keys from short-term (or session) keys?

Por vários motivos, como as chaves de longa duração podem ser passwords e estas normalmente são difíceis de memorizar, convém protegê-la ao máximo e usá-la poucas vezes, ou seja, não são usadas para proteger dados, mas sim para fazer wrapping a chaves de curta duração. E outro motivo é que embora estas chaves também possam ser simétricas, na maioria dos casos elas são assimétricas pois como vimos na week 10 as chaves públicas não são tão eficientes em cifrar e como não as iremos usar muitas vezes nem para cifrar textos longos parecem ideal porque também garante assinaturas e facilitam a troca de chaves que depois levará ao aparecimento das chaves de curta duração.

As chaves de curta duração são sempre chaves simétricas e são para serem descartadas a cada final de sessão de forma a que se forem comprometidas apenas uma porção de dados, num determinado período de tempo, é também comprometida.

E outra vantagem adicional para este esquema é que se eventualmente as chaves de longa duração forem também corrompidas, devido a usarmos chaves de curta duração descartáveis, não é possível comprometer as mensagens passadas porque implicaria conhecer essas chaves de sessão que foram perdidas.

2 - Which ways can one use to generate short-term keys? What guarantees should they give?

Chaves de curta duração ou chaves de sessão podem ser geradas de 3 formas:

1) Podem ser derivadas de outras chaves, normalmente de uma chave de maior duração.

2) Através de um protocolo de acordo de chaves, onde duas chaves de longa duração são pré-acordadas previamente e onde consequentemente são usadas para realizar um “handshake” de forma a que ambos derivem uma chave de sessão.
3) Podem ser distribuídas através de um agente de confiança, onde cada entidade fala com o agente através de uma chave pré-estabelecida, esse agente é responsável por guardar as chaves de todas as entidades e gera chaves de sessão para que essas entidades possam comunicar entre si, ou então com as chaves que já possui cria um túnel onde o agente passa a ser o middle man.

3 - What is a man-in-the-middle attack?

Este tipo de ataque consiste num invasor que está a escutar nos canais da rede de forma secreta e que possivelmente altera as comunicações entre as duas partes. O exemplo mais clássico é conhecido por "eavesdropping", no qual o atacante faz conexões independentes com as vítimas e retransmite mensagens entre elas para fazê-las acreditar que estão falando diretamente entre si por meio de uma conexão privada, quando na verdade toda a conversa é controlada pelo atacante.

4 - How does authenticated Diffie-Hellman avoid man-in-the-middle attacks?

Dotando ambas as entidades participantes da comunicação com um par de chaves (chave pública e chave privada), onde diferente do DH não autenticado agora as entidades depois de passarem o A e o B e as respectivas chaves públicas de cada um, no qual designamos este processo como “trace”, cada entidade depois cria uma assinatura digital sobre o trace com a sua chave privada e envia para a outra entidade para que possa ser validada junto com o trace e a assinatura pública da entidade que emitiu a respectiva assinatura digital. Este processo garante que ambos estão a falar com a entidade que pretendem (autenticação) porque o atacante que realiza o man-in-the-middle não consegue forjar a assinatura do trace de nenhuma entidade pois não sabe as suas chaves privadas, garantindo assim uma autenticação mútua.

Guided practical assignment

1 - Use openSSL to generate Diffie-Hellman parameters at 128-bit security (4096-bit modulus) using option `dhparam`. Do not activate option `-dsaparam`.

$ openssl dhparam 2048

2 - Repeat the exercise activating option `-dsaparam`.

$ openssl dhparam -dsaparam 2048

3 - Why does the first approach take so much longer? Use Sage to check that the produced primes have the structure you describe in your answer.

Porque quando esta opção é utilizada, é utilizado parâmetros de DSA (Digital Signature Algorithm) ao invés de parâmetros DH (Diffie-Hellman). Gerar estes primos é mais fácil e torna o tamanho do expoente mais pequeno o que torna esta operação mais eficiente. O problema é que estes números primos não são tão seguros pois fazem parte de um pequeno subgrupo que pode ser usado por um atacante, daí o aconselhável ser usar esta chave DH, com estes parâmetros, apenas uma única vez.

4 - Use to check that DH works for both parameter sets:

- generate exponents x,y in the range[0. . . q[where q is the order of the group generator
- computeX=g^$_x$ (mod p) and Y=g^$_y$(mod p)
- check that X^$_y$ (mod p) = Y^$_x$ (mod p)