INE 5680- Segurança da Informação e de Redes - Prova 1 – 17-10-2014 – Turma A e B
Prof. Bosco Aluno: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

PARTE I – Questões para indicar Verdade ou Falso.

1. (Vulnerabilidades, Ameaças, Risco e Ataque)

1.1. (Verdade/Falso) Um email recebido portando um link suspeito, mas que é não é aberto, constitui uma ameaça, mas se aberto, dependendo do que existe no link, pode ser um ataque por software, via algum programa de vírus, cavalo de tróia ou um *worm*.

Verdade. Se você não clicou no link, nada irá acontecer. O ataque só se concretiza caso você clique.

1.2 (Verdade/Falso) Ameaça é o grau de severidade de um ataque.

 Falso. Ameaça é a tentativa de um ataque.

2. (Criptografia Simétrica)

1. (Verdade/Falso) Para um algoritmo de criptografia simétrica ser computacionalmente seguro, um

 dos seguintes critérios é atendido: (a) o custo para quebrar a cifra é muito maior do que valor da
 informação cifrada; (b) o tempo exigido para quebrar a cifra é muito maior ao tempo de vida útil da
 informação.

 Verdade. Estas são boas razões para aceitarmos/confiarmos em algoritmos de criptografia simétrica
 E por isso, utilizamos, até hoje, algoritmos antigos que já foram quebrados, como o caso do uso do
 DES.

1. (Verdade/Falso) O modo de cifra EBC (cifra blocos um-a-um), mais apropriado para aplicações com pequenas mensagens para serem criptografados para determinadas aplicações, é mais seguro que o modo de cifra CBC (cifra blocos com encadeamento de blocos) mais apropriado para aplicações com pequenas mensagens para serem criptografados.

Falso. CBC é mais seguro do que EBC, porque CBC faz o encadeamento usando os blocos e EBC faz a criptografia de blocos, um-a-um. Mas, EBC serve para aplicações com pequena informação a ser protegida.

1. (Verdade/Falso) Qualquer mensagem pode ser criptografada com o algoritmo do acordo de Diffie-Hellman.

Falso. Diffie-Hellman não criptografa, apenas resolve o acordo de uso, por ambas as partes se comunicando, de uma chave compartilhada.
2. (Verdade/Falso) Chaves mestras servem para comunicar usuário de um sistema com um servidor de serviço. Chaves de sessão servem para criptografar dados entre componentes de um protocolo de cifra simétrica.

Estas são as diferenças conceituais entre chaves mestras e chaves de sessões.

1. (Hash)
2. (Verdade/Falso) Precisa ser inviável computacionalmente reencontrar a mensagem de entrada baseada apenas em seu hash.

Uma função Hash não tem, matematicamente, sua função inversa correspondente.
3. (Verdade/Falso) HASH é uma técnica de criptografia que visa ocultar o texto cifrado.

Apesar de Hash aparecer junto ao tema sobre criptografia, uma função Hash fornece um representante da mensagem dada, num tamanho fixo, que depende da construção da função.
Criptografia moderna, ainda é, criptografia simétrica e criptografia assimétrica.
4. (Verdade/Falso) Matematicamente é possível, mas não é conveniente ser computacionalmente possível, achar duas mensagens distintas como o mesmo *hash*.

O espaço de resultados *(2­En)* para uma função *Hash* de tamanho *n*, é um conjunto grande, mas é finito. Por isto, matematicamente, pode haver colisões entre cálculos de hashes, mas isto não é conveniente em segurança.
5. (Verdade/Falso) Deve ser possível achar uma mensagem particular que tenha um *hash* específico.

Se fosse assim, para que existir esta função Hash. A função não tem a sua função inversa correspondente.

4. (Distribuição de Chave Pública)

(Verdade/Falso) Várias técnicas tem sido propostas para distribuição de chaves públicas. Praticamente todas essas propostas podem ser agrupadas nos seguintes esquemas gerais: (a) anúncio público de chaves públicas, (b) diretórios distribuídos disponíveis publicamente, (c) autoridade de chave pública e (d) certificados de chave pública. O GnuPG gerencia a distribuição de chaves públicas através do anúncio de chaves públicas.

Falso. GnuPG, como mostrado na aula prática sobre geração de chaves e segurança de email, usa diretórios distribuídos disponíveis publicamente, que são inclusive, replicados em servidores na Internet.

5. (Assinaturas e Certificados)

(Verdade/Falso) Considere se o algoritmo Diffie-Hellman para geração de uma chave compartilhada na Internet. Os participantes no algoritmo de Diffie-Hellman, no início do algoritmo, não se autenticam e, portanto, este pode ser um problema no uso do algoritmo, entre dois participantes na Internet.

Realmente, só trocam números iniciais, mas sem se autenticarem. Basta ver o algoritmo.

(Verdade/Falso) Assinatura digital por criptografia de chave pública resolve o requisito de segurança de confidencialidade.

Autenticidade, Integridade e Não-repúdio.

(Verdade/Falso) Assinaturas digitais, por si só, servem muito bem à verificação de uma quantidade limitada de pessoas, com as quais você está familiarizado (por exemplo, ambiente da UFSC).

Porque fica fácil de se acreditar de quem é mesmo a chave pública enviada.

(Verdade/Falso) Mas, se você está na Internet, isto não significa muita coisa. Afinal, a chave pública enviada pode ser capturada por alguém, e esta pessoa poderia estar se passando por outra.

Obvio. Para quem conhece a Internet, como um meio não confiável.

(Verdade/Falso) Na Internet, você pode saber, que quem envia é quem diz ser, pelo esquema da assinatura digital.

Já explicado.

 (Verdade/Falso) Uma assinatura estabelece a identidade do criador legal de um documento, mas você pode obter duas assinaturas idênticas com uma mesma chave privada e para dois documentos distintos.

Porque a assinatura depende do Hash da mensagem, se muda a mensagem, os hashes são diferentes. Como consequência, as assinatura serão distintas.

(Verdade/Falso) As assinaturas podem ser usadas em documentos, estejam criptografados, ou não. O verdadeiro valor das assinaturas é que elas inequivocamente identificam o criador do documento e detectam se esse foi alterado ou não.

Sim, autentica e verifica a integridade.

(Verdade/Falso) Devido ao fato de não ser computacionalmente viável forjar uma assinatura, sem a posse da chave de assinatura, o autor não pode repudiar o fato que assinou uma mensagem.

A chave de assinatura é a chave privada, com a qual o autor assina. Se a verificação é feita com a chave pública, então o autor não pode repudiar.

(Verdade/Falso) Um certificado digital é uma coleção de dados que contém, entre outras coisas, o nome de uma entidade que está sendo autenticada (o desconhecido), o nome de um assinante do certificado (geralmente o nome de uma AC), a chave pública da entidade que está sendo certificada e a assinatura da entidade confiável (AC). Um certificado autentica seu proprietário.

Obvio. O certificado contém uma chave pública fornecida por uma AC confiável.

(Verdade/Falso) Existem diferentes tipos de certificados: certificados de email, certificados de servidor, certificados de clientes (e aqui existem alguns outros tipos, dependendo para que servirá o certificado). Mas, se você tem uma aplicação e deseja testar esta, usando certificados, você pode gerar, para o ambiente de sua aplicação (email, VPN, ou outra) um tipo de certificado que só serve para o fim de testes, que é chamado certificado auto-assinado.

Quando se faz uma tarefa de VPN com criptografia assimétrica dá para se gerar certificados auto-assinados., que representarão os certificados de servidores e os certificados de clientes.

**PARTE II - SOBRE NÃO-REPÚDIO** (1,0)

2. Responda as questões, explicando sua resposta.Alice *(A)* assina digitalmente uma mensagem *M* para Bob, usando sua chave privada *KRA* sobre o *H(M)* com um algoritmo de assinatura (como o do RSA). Bob obtém a chave pública *KUA* relacionada a *KRA*.

1. Como o requisito de segurança de ***não-repúdio***, por parte de Alice, é garantido pela verificação da assinatura ? (0,5)

Uma assinatura digital garante não-repúdio, porque uma vez que Alice criptografe o *hash* da mensagem M com sua chave privada KRA, somente a chave pública KUA correspondente da KRA pode verificar a assinatura, o que garante que foi Alice quem enviou a mensagem M.

Ou com outras palavras, devido ao fato de não ser computacionalmente viável forjar uma assinatura sem a posse da chave de assinatura (chave privada), o autor da assinatura não pode repudiar o fato que assinou uma mensagem.

1. Suponha que Alice revele acidentalmente sua chave privada *KRA*para um terceiro. Como o ***não-repúdio*** pode ser garantido neste caso ? (0,5)

A revelação da chave privada de Alice, KRA, acidentalmente ou maliciosamente, acaba impedindo a garantia de se saber quem enviou, de fato, a mensagem, isto é, neste caso, o não-repúdio não é garantido.

**PARTE III - PROTOCOLOS CRIPTOGRÁFICOS**
**A) NONCES (1,0)**

Existem três formas típicas de usar *nonces* como desafios. Suponha que Na seja um *nonce* gerado por A, A e B compartilham a chave K, e f() seja uma função como incremento. Os três casos são:

**Uso 1**(1) A 🡪 B:Na

(2) B 🡪 A: E(K,Na)

**Uso 2**
(1) A 🡪 B: E(K,Na)

(2) B 🡪 A: Na

**Uso 3**(1) A 🡪 B: E(K,Na)
(2) B 🡪 A: E(K,f(Na))

1. (Verdade/Falso) Todas as três servem ao esquema proposto. A diferença está na vulnerabilidade. (0,20)
2. (Verdade/Falso) No Uso 1, um ataque poderia violar a segurança capturando *Na* e retendo uma resposta de B para um futuro ataque por repetição. (0,20)
3. (Verdade/Falso) O atacante poderia tentar prever uma possível repetição no Uso 2, mas isso não daria certo se os *nonces* forem aleatórios. (0,20)
4. (Verdade/Falso) Tanto no Uso 1 como no Uso 2, as mensagens trabalham em ambas as direções. Ou seja, se N é enviada em uma direção, a resposta é E[K, N]. (0,20)
5. (Verdade/Falso) No Uso 3, a mensagem é criptografada em ambas as direções; o propósito da função é assegurar que as mensagens 1 e 2 são idênticas. Então, o Uso 3 é mais seguro. (0,20)

Comentário: a) (Verdade) Todas as três servem ao esquema proposto. A diferença está na vulnerabilidade. b) (Verdade) No Uso 1, um ataque poderia violar a segurança capturando *Na* e retendo uma resposta de B para um futuro ataque por repetição, uma forma de ataque *suppress-replay (definido na lista)*. c) (Verdade) O atacante poderia tentar prever uma possível repetição no Uso 2, mas isso não daria certo se os *nonces* forem aleatórios. d) (Falso) Tanto no Uso 1 como no 2, as mensagens trabalham em ambas as direções. Ou seja, se *Na* é enviada em uma direção, a resposta é *E[K, Na]*. Vide o caso do Uso 2, onde a resposta é *Na* a descoberto. e) (Falso) No Uso 3, a mensagem é criptografada em ambas as direções; o propósito da função é assegurar que as mensagens 1 e 2 não são idênticas. Então, o Uso 3 é mais seguro.

**B) INTEGRIDADE E AUTENTICAÇÃO (1,5)**

1. Seja um sistema de *shopping center* B2B, de comércio eletrônico, na Internet, com as seguintes características e requisitos:

- Empresas vendedoras V acessam o sistema *shopping* e oferecem seus
 produtos.

- Pessoas compradoras P acessam o sistema shopping para consulta a preços.

- Pessoas compradoras P acessam o sistema e realizam pedidos.

**a)** Que ataques que podem ocorrer para pessoas compradoras P ou empresas vendedoras V, nesse ambiente B2B ??? (0,50)

Resposta:

 ***Ataques a empresas vendedoras V:***

1. Autenticação: algum site quer se fazer passar pela empresa real, fazendo ofertas falsas.
2. Integridade: algum atacante quer modificar a oferta da empresa.

 ***Ataques a pessoas compradoras:***

1. Autenticação: alguém quer se fazer passar por alguém, fazendo compras falsas.
2. Integridade: alguém pode modificar a compra de outra pessoa.

**b)** O seguinte protocolos criptográficos podem mitigar esses problemas de ataques ? Se sim, mantenha os protocolos. Caso não, corrija os erros dos protocolos. (0,5)

Protocolo de geração de oferta e verificação, se empresa V é legítima:

**V : Empresa Vendedora**

 V : Gera Oferta

V 🡪 B2B : S***KUv*** ( Oferta) || *KUv* // V faz oferta no site e assina
 // oferta
 S***KRv*** ( Oferta) || *KUv* // a assinatura é feita pela chave privada.

 B2B : *KRv* ??? // verifica a se empresa é cadastrada

 B2B : *KUv* // a assinatura é verificada pela chave pública.

 B2B : V***KRv***( Oferta ) // verifica originalidade da oferta

 B2B : Armazena Oferta

Protocolo de consulta de oferta e geração do pedido:

**P : Pessoa Compradora**

P 🡪 B2B : Consulta Oferta

 P : Gera Pedido

P 🡪 B2B : S**KRc** ( Pedido) || KUc // assina pedido

 B2B : KUc ??? // verifica se empresa cadastrada

 B2B : V**KRc** ( Pedido ) // verifica originalidade do pedido

 B2B : V**KUc** ( Pedido ) // verifica originalidade do pedido

 B2B: V**KUc**

 B2B : Armazena Pedido

Resposta: Não, ele possui erros.

c) Descreva um outro protocolo de geração de oferta e verificação se empresa V é legítima mas agora, utilizando uma estrutura de chave pública e certificado digital. (0,5)

**V : Empresa Vendedora**

 V : Gera Oferta

V 🡪 B2B : E***KU-B2B*** ( Oferta) || C(*KUv*) // V faz oferta no site,
 criptografa oferta e concatena seu certificado com sua *KUV*

 B2B : C(*KUv)* ??? // valida o certificado de V, na AC, para ver
 se a empresa é cadastrada

 B2B : D***KR-B2B***( Oferta) // decriptografa a oferta

 B2B : Armazena Oferta

**C) ACESSO SEGURO À BANCO DE DADOS (1,5)**

O Detran de Florianópolis descobriu que seu sistema de Banco de Dados BD, para emissão de documentos para automóveis está apresentando problemas de segurança. O Banco de Dados BD possui um par de chaves assimétricas *(KUBD, KRBD)* que pode ser usado para a segurança de seu acesso. E o funcionário possui um par (KUF , KRF).

Sabendo-se que o *login* ao sistema é realizado informando o número do CPF do funcionário e uma senha de acesso, aponte possíveis vulnerabilidades e proponha um protocolo criptográfico para solucionar o problema apresentado abaixo:

Problema: Foram detectados acessos e uso do sistema por CPFs, cujo funcionário titular nega ter
 realizado.

Componentes do protocolo: F é o funcionário que faz acesso. BD é o nome do sistema acessado. S é a senha de acesso ao BD.

Vulnerabilidades possíveis: : O CPF é número público e não tem necessidade de se manter sigilo. O problema deve estar na senha, por ter sido capturada.

F 🡪 BD : CPF
BD 🡪 F : requisição da senha S
F 🡪 BD : E ( KUBD , S) )
 BD : D (KRBD, S) )
 BD : Hash(S) // BD autentica U
BD 🡪 F : <mensagem permitindo acesso>
F 🡪 BD : E (KUBD , <requisição de documento de veículo>)
 BD : D (KRBD, <requisição de documento de veículo>)
BD 🡪 F : E (KUF , < documento de veículo>)
 F: D (KRF , < documento de veículo>)
 F: < documento de veículo>

Ou outra solução proposta.

1. **ELEIÇÕES COMPUTADORIZADAS SEGURAS (2,0)**

Votação computadorizada nunca pode ser usada para eleições gerais, a menos que, exista um protocolo que mantenha a privacidade individual e proíba fraudes. O protocolo ideal deve ter, pelo menos, estes seis requisitos:

1. Somente eleitores autorizados podem votar.
2. Ninguém pode votar mais de 1 vez.
3. Ninguém pode determinar para quem, um eleitor deve votar.
4. Ninguém pode duplicar votos.
5. Ninguém pode mudar o voto de alguém, sem ser descoberto.
6. Todo eleitor pode ter certeza que seu voto foi levado em consideração na apuração final.

Suponha o TSE brasileiro, ser a Central de Apuração Final (CAF) e considere o seguinte protocolo de votação computadorizada, para cada eleitor:

[1] Cada eleitor assina seu voto com sua chave privada.
[2] Cada eleitor encripta seu voto assinado com a chave pública do TSE.
[3] Cada eleitor envia seu voto, seguro, ao TSE.
[4] O TSE decripta o seu voto, verifica a assinatura, tabula seu voto, e publica os resultados.

Faça um análise deste protocolo e responda, explicando suas respostas, as seguintes questões:

**a)** Quais requisitos acima (numerados de 1 a 6) são satisfeitos por este protocolo ??? Responda se o requisito é satisfeito e explique suas respostas.

1. Somente eleitores autorizados podem votar.

Satisfaz 1. Porque o cada eleitor é autenticado por sua assinatura. Cada voto é assinado com a chave privada do eleitor, e assim, o TSE sabe quem votou e quem não votou, e com que frequência cada eleitor votou.
2. Ninguém pode votar mais de 1 vez.

Satisfaz 2. O TSE registra votos recebidos na etapa [3]. O número de registro de cada eleitor no TSE deve ser único e seu par de chaves (pública e privada) também.

1. Ninguém pode determinar para quem, um eleitor deve votar.

Dificil satisfazer 3. Ninguém pode mudar qualquer voto de algum outro eleitor, mesmo se ele é interceptado na etapa [3].

1. Ninguém pode duplicar votos.

Satifaz 4. Se um voto aparece, em que não é assinado por um eleitor autenticado, ou se um segundo voto vem assinado por um eleitor que já votou, o TSE ignora.
2. Ninguém pode mudar o voto de alguém, sem ser descoberto.

Não satisfaz 5. Por (4), O TSE ignora, não tem como descobrir que votou em duplicidade.
3. Todo eleitor pode ter certeza que seu voto foi levado em consideração na apuração final.

O eleitor não pode ter certeza de que seu voto foi computado, pelo fato de ser tabulado, pois om protocolo, não indica como é feita esta tabulação de votos.

**b)** De modo geral, quais problemas, se existem, neste protocolo ?

Resposta: O problema neste protocolo é que a sua assinatura é anexada ao seu voto. Assim, o TSE, saberá quem votou para quem, quebrando o sigilo do voto. Encriptando seu voto com a chave pública do TSE, isto previne (impede) qualquer um de “escutar” o protocolo e descobrir quem votou para quem. Entretanto, o eleitor tem de confiar completamente no TSE, e isto é, análogo a ter um juiz eleitoral olhando por cima da cabine de votação.

Algo parecido com o que acontece na apuração final do TSE brasileiro, pois a apuração final não é transparente para os eleitores. Outro detalhe da nossa eleição é que quando o eleitor comparece para votar e a urna é ativada para ele, esta é ativada com os números do eleitor e não é visível ao eleitor o que acontece com os números dele. Além, do eleitor, não ter certeza se seu voto dado, realmente, foi dado ao candidato eleito por ele. Mais detalhes, você pode encontrar no site [www.vocefiscal.org.br](file:///C%3A%5CUsers%5CJoao%20Bosco%20M.%20Sobral%5CDocuments%5Cine5680%5C2014-2%5CProva%201-17102014%5Cwww.vocefiscal.org.br)
ou no links de vídeo que encaminhei à turma.

**c)** Indique uma conclusão geral sobre este protocolo.

O protocolo mostra como é difícil alcançar, já, os primeiros três requisitos (1, 2 e 3) de um protocolo de votação segura, e muito menos os outros. O protocolo é insuficiente quanto a sua segurança. Existem outros mais seguros, mas não 100% seguros.