Criptografia Aplicada 2006/2007
Enunciado do Trabalho Prático

Manuel Bernardo Barbosa – mbb@di.uminho.pt
José Bacelar Almeida – jba@di.uminho.pt

October 3, 2006

1 Introdução

As aulas práticas de Criptografia Aplicada serão preenchidas com a elaboração de pequenos projectos de programação, por grupos de até três alunos.

O objectivo final das aulas práticas é a integração dos componentes desenvolvidos ao longo do semestre numa aplicação demonstrativa das tecnologias abordadas no âmbito da disciplina.

Os projectos serão implementados na linguagem JAVA, utilizando as seguintes APIs especializadas em funcionalidades de segurança:

- Java Cryptography Architecture (JCA) – API básica para integração de funcionalidade de segurança em aplicações JAVA.
- Java Cryptography Extension (JCE) – extensão à JCA que inclui técnicas criptográficas mais poderosas.
- APIs utilitárias para manipulação de estruturas de dados ASN.1.

A JCA é um componente básico da infra-estrutura JAVA distribuída pela SUN (JDK). Esta infra-estrutura inclui também a implementação da JCE da própria SUN. Conjuntamente, estas duas APIs fornecem implementações dos algoritmos criptográficos mais comuns. No entanto, recomenda-se a utilização da implementação JCE da IAIK que, para além de mais completa, está integrada de forma elegante com a API para manipulação de estruturas de dados ASN.1, do mesmo fabricante. A instalação básica do JDK apresenta uma limitação das políticas de utilização de algoritmos criptográficos que tem de ser desbloqueada substituindo os ficheiros correspondentes por versões mais liberais. Estes ficheiros podem ser descarregados do site da SUN.

2 Instalação das APIs

A instalação deve ser feita do seguinte modo (versão 1.4.2):

```
1 - Instalar o Java
2 - Alterar a política de segurança, expandindo o ficheiro
                       jce_policy
fornecido pela SUN na directoria
                $JAVA_PATH/jre/lib/security
3 - Instalar a biblioteca IAIK, que pode ser obtida em
             http://jce.iaik.tugraz.at/
4 - Copiar o ficheiro iaik_full.jar  para a directoria
                $JAVA_PATH/jre/lib/ext
5 -  Instalar o provider IAIK inserindo no ficheiro
         $JAVA_PATH/jre/lib/security/java.security
a seguinte linha
        security.provider.7=iaik.security.provider.IAIK
```

3 Objectivos do Trabalho Prático

Pretende-se construir um sistema de votação electrónica para uma comunidade fechada, como uma empresa. As votações em causa terão como universo de votantes um sub-conjunto dos membros da comunidade (e.g. votações restritas à administração). Dada a complexidade do problema da votação electrónica remota, optou-se por um protocolo de votação simplificado que, não permitindo alcançar todos os requisitos de segurança expectáveis de um sistema para eleições em larga escala, apresenta propriedades aceitáveis neste contexto.

3.1 Descrição do processo de votação

O sistema de votação é composto pelos seguintes componentes:

- **Administrador** — entidade que estabelece quem está habilitado a participar na votação e que controla o desenrolar do processo de votação no período estabelecido.
- **Contador** — entidade responsável por realizar a contagem dos votos após a realização da eleição.
- **Votante** — membro da comunidade que está habilitado a votar na eleição.

A distinção entre as entidades *Administrador* e *Contador* destina-se a garantir o anonimato dos votos: o administrador sabe a identidade de quem vota, mas não o sentido desses votos; enquanto o contador saberá o sentido dos votos mas nunca a identidade dos votantes. Assume-se que essas entidades nunca irão cooperar no sentido de comprometer o anonimato dos votos.

O protocolo de votação processa-se ao longo de três fases, como se descreve abaixo:

Fase I: Inicialização

O administrador A publica os dados relativos à votação a realizar. Em particular, é publicada uma lista com a identificação dos votantes autorizados e qual o período de tempo durante o qual se realizará a votação. Aos votantes autorizados, fornece ainda os parâmetros que permitem efectuar a lacragem dos seus votos de forma a que apenas o contador os possa recuperar.

Fase II: Votação

- Um votante autorizado V acede ao administrador A.
- A só aceita a conecção uma vez confirmada a identidade de V, e verificada a sua inclusão na lista de votantes autorizados.
- V produz o boletim de voto. Este é composto pelo sentido de voto propriamente dito, e por um *nonce* n (número aleatório). Deve guardar de forma segura n para verificação futura.
- V lacra o voto por forma a que apenas o contador o possa recuperar. O voto lacrado é enviado por canal seguro para o administrador.
- A aceita o voto lacrado se o votante estiver autorizado e se ainda não exerceu o seu direito de voto (i.e. ainda não submeteu um voto lacrado). Devolve ao votante um recibo em que certifica a recepção do voto lacrado.

Fase III: Contagem dos votos

- A publica duas listas não associáveis (i.e. a ordem dos elementos em cada uma delas é completamente aleatória) com os votantes que exerceram o direito de voto e os boletins lacrados.
- C acede à lista de boletins lacrados e recupera o conteúdo de cada um dos boletim de voto.
- C publica os resultados da votação numa lista contendo os conteúdos de todos os boletins aceites (sentido de voto e o *nonce* associado), mais uma vez numa lista com ordem aleatória.
- Cada votante verifica se o seu voto foi correctamente contabilizado com base no *nonce* n que só ele conhece (gerado na fase de votação). Em caso de erro (voto lacrado não aparece, o *nonce* não aparece, ou aparece associado a um voto errado) o votante pode impugnar as eleições com base no recibo emitido por A.

3.2 Implementação

Durante as aulas teórico-práticas será discutida a seguinte arquitectura para este sistema.

- **Servidor WEB da intranet** O servidor WEB será utilizado como difusor de informação nos pontos em que o processo de votação prevê a sua publicação (e.g. a lista dos votantes autorizados). O principal requisito que se colocará sobre este servidor é o controlo de acessos — só as entidades envolvidas num dado ponto deverão ter acesso à informação produzida aí (e.g. só os votantes autorizados deverão ter acesso às listas produzidas no processo de uma votação). Mesmo sendo concebível um sistema que envolvesse a geração dinâmica do conteúdo desse *site*, podemos para simplificar considerar um *site* estático onde os ficheiros de informação são introduzidos manualmente ao longo do processo.

- **Servidor e Clientes para a votação** Durante o período de votação, deverá ficar activo um *deamon* (executado pelo *administrador*) que será responsável por coleccionar os votos dos vários votantes. O voto será submetido com o auxilio de uma aplicação cliente, responsável pela interacção com o *deamon* referido acima.

- **Contador** Para a contagem dos votos, deverá ser considerado um pequeno programa java que realize as tarefas associadas.

Esta arquitectura estará inicialmente reduzida uma infra-estrutura básica de comunicação, e evoluirá através da incorporação gradual de técnicas e protocolos criptográficos apresentados nas aulas teóricas.

Algumas das técnicas que deverão ser utilizadas na elaboração dos trabalhos práticos são as seguintes:

- Servidor HTTPS com configuração avançada do modo SSL, incluindo especificação de algoritmos e configuração de certificados X.509 de servidor e de cliente.

- Cifras simétricas e MACs.

- Cifras assimétricas e assinaturas digitais.

- Protocolos de acordo de chaves.

- Pedidos de certificado PKCS#11 e armazenamento de credenciais PKCS#12.

A identificação dos requisitos de segurança inerentes a cada tipo de interacção, e a sua correcta implementação, serão também discutidas nas aulas da disciplina, e serão o factor central na avaliação dos trabalhos práticos.

4 Avaliação

A avaliação do trabalho prático basear-se-á num relatório do trabalho efectuado, a entregar no final do semestre. Na altura da entrega do relatório terá lugar uma apresentação/demonstração da aplicação desenvolvida, sujeita também a avaliação.