Segurança Informática e nas Organizações 2015-16

Projeto: Mecanismos de Proteção de Software

10 de Setembro de 2015 V 1.0

1 Introdução

Um aspeto muito importante do desenvolvimento de aplicações é o controlo de integridade e o controlo de execução das aplicações desenvolvidas. Considerando que qualquer desenvolvimento possui um custo não desprezável, constituindo o meio de vida dos programadores, é simples considerar que muitos autores pretendam disponibilizar a aplicações por si desenvolvidas de forma não gratuita.

Considerando esta situação, é comum as aplicações incorporarem mecanismos destinados a comprovar que o utilizador tem de facto uma cópia autorizada da aplicação, e que a cópia encontra-se a executar de acordo com os parâmetros acordados entre o autor e o utilizador. Um parâmetro comum é a restrição de execução num número limitado de dispositivos.

O processo pelo qual é realizado este controlo tem as suas raízes na utilização de cifras simétricas e assimétricas, assim como funções de síntese e assinaturas digitais. O princípio básico consiste na criação de uma licença que não é mais do que uma assinatura criada pelo autor ou distribuidor, algo relacionada com a aplicação em execução. É comum considerar-se o nome da aplicação e a sua versão, podendo igualmente considerar-se um identificador do sistema em que a aplicação executa ou mesmo uma síntese do binário em execução.

Também é comum utilizarem-se dispositivos físicos, frequentemente denominados por *dongles* ou chaves de *hardware*, que restringem a execução aos sistemas onde aquele dispositivo se encontre conectado.

1.1 Objetivo

O objetivo deste trabalho é o de desenvolver um sistema denominado que permita distribuir aplicações de forma segura, garantindo que apenas são executadas por donos legítimos das mesmas.

Este sistema deverá ser composto por um conjunto de ferramentas que executam nas instalações do autor ou distribuidor da aplicação, e por uma biblioteca que é incorporada nas aplicações. Na concretização deste trabalho os alunos devem considerar uma qualquer aplicação desenvolvida pelos mesmos ou por outros, numa qualquer linguagem de programação da sua preferência.

As ferramentas disponíveis ao autor permitem criar ficheiros de licenças que são fornecidos às aplicações, com uma especificação de execução e identificação dos dados da licença. Estes ficheiros serão descritos na subsection 1.2.

A biblioteca deverá ser constituída por um módulo contendo um conjunto de funções (ou métodos numa classe). É vital que esta biblioteca seja completamente independente da aplicação desenvolvida, no máximo partilhando a linguagem de programação.

Esta biblioteca deverá possuir o seguinte conjunto de métodos:

- `void init(string nomeDaApp, string versão)`: Esta função corresponde à inicialização da biblioteca de controlo de execução. No caso de uma implementação orientada a objetos, esta função pode existir ou corresponder ao construtor.

- `bool isRegistered()`: Uma aplicação deverá invocar esta função no início da sua execução e sempre que ache necessário. Ela deverá executar de forma rápida e eficiente, validando a correta execução da aplicação atual. Caso se verifique que a aplicação executa de forma autorizada, ela não deverá imprimir qualquer valor e deverá devolver o valor `True`. Caso contrário esta deverá devolver o valor `False`

- `bool startRegistration()`: Esta função deve apresentar um interface (da forma mais adequada à aplicação, o que pode utilizar o `stdout` ou um interface gráfico) indicando que a aplicação não se encontra registada e possibilitando iniciar o processo de registo de uma nova licença. Os detalhes sobre este processo encontram-se na subsection 1.4.

- `void showLicenseInfo()`: Esta função apresenta os dados da licença atual (caso ela exista), ou informação de que a aplicação não se encontra registada. A apresentação desta informação, mais uma vez, deverá ser efetuada da maneira mais adequada à aplicação. Para uma aplicação de linha de comandos esta informação pode ser escrita para o terminal (`stdout`).

1.2 Ficheiros de Licença

Os ficheiros de licenças de software deverão conter diversa informação que permita identificar o utilizador autorizado e o ambiente de execução da máquina. É deixado ao critério dos alunos a escolha da informação, sugerindo-se pelo menos a seguinte:

- informação que identifique o utilizador: o seu nome, endereço de email, número de identificação e certificado de chave pública do Cartão de Cidadão.

- informação que identifique o sistema: um identificador do sistema obtido do conjunto de hardware presente tais como número e tipo CPUs, placas de rede (endereços MAC), números de série do suporte de armazenamento, ou mesmo identificadores da BIOS.

- informação que identifique a aplicação: nome da aplicação, versão atual, valor da síntese do seu ficheiro principal (ou ficheiros relevantes) e, se a biblioteca existir de forma separada, qual o valor da síntese do ficheiro contendo a biblioteca.

- informação que identifique o intervalo temporal: data de início de validade da licença e data de expiração.

Toda esta informação deverá depois ser cifrada, sendo adicionado um controlo de integridade sobre a forma de uma assinatura efetuada pelo autor ou distribuidor. Devem ser utilizadas chaves simétricas e assimétricas da forma mais adequada.

1.3 Validação da Licença

Validar uma licença implica vários passos, verificando primeiramente o próprio ficheiro de licença e depois os diferentes componentes que ela codifica.

O primeiro passo consiste em validar a assinatura da mesma e decifrar o seu conteúdo (por esta ordem ou pela inversa). Pode-se assumir que cada aplicação distribuída contém uma chave que fornece à biblioteca de proteção a quando da sua inicialização. Pode igualmente considerar que a biblioteca deriva uma chave a partir dos dados da máquina ou aplicação, ou mesmo que existe uma chave pré-distribuída na biblioteca.

A informação relativa ao intervalo temporal de execução deve ser validado considerando a data atual do sistema.

A informação relativa à identificação do sistema deve ser calculada na inicialização da biblioteca e confrontada com a informação presente no ficheiro de licença. A biblioteca deve ser capaz de tolerar pequenas alterações ao sistema, o que é comum caso exista a troca de um CPU ou de uma placa de rede. Após excedida esta tolerância, considera-se que o sistema não é mais válido.

A informação relativa à identificação do utilizador é útil na medida que torna o Cartão de Cidadão num dispositivo de segurança, condicionando a execução à presença de um cartão específico, com uma chave privada específica. A biblioteca, fazendo uso da chave pública registada na licença poderá autenticar o utilizador. Visto que esta operação é mais demorada que as anteriores, deixa-se ao critério dos alunos decidirem quando é que o cartão é validado. Depois de corretamente validado, pode-se considerar que testar a presença do cartão, sem realizar assinaturas é suficiente para continuar a execução.

A informação relativa à identificação da aplicação serve para validar a integridade do sistema. Garante-se desta forma que a aplicação não foi manipulada (de forma simples), numa tentativa de ignorar o sistema de validação de licença.

1.4 Registo

O processo de registo implica a biblioteca recolher toda a informação necessária (utilizador, sistema e aplicação), o que é expresso num documento (e.x., codificado em base64). Este documento deve ser assinado pela chave presente no Cartão de Cidadão do utilizador em causa, sendo depois cifrado de forma a que forme um pedido de registo seguro. O documento de pedido de registo pode ser simplesmente apresentado no ecrã, sendo responsabilidade do utilizador a sua cópia e envio. Em alternativa pode igualmente ser escrito

no sistema de ficheiros.

O autor terá de utilizar um conjunto de ferramentas criadas para o efeito de processar o pedido de registo, emitindo um ficheiro de licença assinado. Considere que o autor possui um par de chaves público e privado e considere a utilização de cifras híbridas. Atenção que a assinatura efetuada pelo utilizador e a sua validade temporal têm de ser verificadas.

A duração de cada licença é definida pelo autor.

É obrigatório que estas ferramentas mantenham informação organizada relativa às aplicações distribuídas, licenças existentes e utilizadores. Embora não seja necessária a utilização de bases de dados relacionais pois uma estrutura de diretórios adequada é suficiente, o seu uso não é desencorajado.

Não se considera necessário implementar qualquer mecanismo de comunicação entre sistemas. Desta forma, e considerando um cenário "real" os ficheiros de pedido de registo e de licença podem ser transferidos por meios alternativos (e.x., email, ftp, dropbox, etc..).

Atenção: Todas as chaves privadas devem ser armazenadas de forma segura e não em claro!

2 Avaliação do Projeto

Os projetos devem ser realizados em grupos de 2 elementos. A nota final dependerá de 3 aspetos:

1. O grau de satisfação dos requisitos expostos neste enunciado. Isto é, quantas das funcionalidades pedidas foram implementadas.

2. O grau de complexidade da solução apresentada. São mais valorizadas soluções simples que conseguem o maior grau de integração de funcionalidades e que melhor satisfazem a experiência dos utentes. É também valorizada a identificação, discussão e proposta de solução de alguma eventual vulnerabilidade na aplicação proposta.

3. A participação individual de cada elemento do grupo. Esta será aferida em discussão oral e em casos extremos pela participação no repositório de código do grupo. **O desconhecimento de quaisquer partes relevantes do projeto apresentado será interpretado como não**

tendo participado na sua realização, ou contribuído de forma relevante para a mesma.

No dia **19 de Outubro de 2015** será necessário entregar um documento, com um limite de **2 (duas) páginas**, especificando:

- A aplicação e linguagem escolhidas
- O formato do ficheiro de licenças
- O formato do ficheiro de pedido de registo
- O método de identificação do computador
- Identificação dos componentes de software que compôem o sistema
- Os objetivos e funcionalidades prevista de cada componente a desenvolver.

Será fornecido feedback sobre esta informação, devendo este ser utilizado para refinar o trabalho. Esta avaliação intermédia contribui com 1 valor para a nota final do trabalho em grupo.

No dia **9 de Novembro de 2015** haverá uma avaliação intermédia com a duração máxima de 10 minutos por grupo e na presença de todos os elementos da turma. Nela, cada grupo deve fazer uma apresentação oral que deverá incidir sobre a **arquitetura total do sistema a concretizar** (como tudo funciona) e incluir:

- quais os módulos a implementar, o seu relacionamento e funções
- quais os métodos de identificação utilizados (utilizador, aplicação, sistema) e como serão implementados.
- qual o formato do ficheiro de licenças e de registo
- detalhe sobre os métodos critográficos aplicados
- o que já foi realizado e aspetos relevantes referentes ao trabalho já realizado;
- qual o trabalho que falta realizar e aspetos relevantes sobre o trabalho a realizar.

Considerando uma distribuição do esforço mais ou menos homogénea ao longo do semestre e uma concretização do projeto, espera-se que na altura desta avaliação intermédia os vários grupos já tenham protótipos funcionais de alguns métodos de identificação, uma ideia clara sobre os ficheiros trocados e um protótipo funcional do processo de criação do ficheiro de registo.

Esta avaliação intermédia contribui com 2 valores para a nota final do trabalho em grupo.

No final do semestre, para além da demonstração final do trabalho, os alunos deverão entregar um relatório da sua realização e fazer uma apresentação oral do trabalho. Esta apresentação será seguida de uma discussão individualizada onde o grupo fará a defesa do seu trabalho. O relatório deverá referir todas as decisões tomadas pelo grupo na realização do projeto e todos os requisitos não cumpridos. O relatório deverá ainda conter imagens devidamente comentadas que evidenciem a correção da solução implementada. As datas de entrega do relatório e de apresentação e discussão do trabalho serão oportunamente indicadas.

2.1 Bónus

Serão atribuídos pontos de bónus a funcionalidades que aumentem de forma interessante e razoável a segurança do sistema. Não se procuram sistemas complexos mas sim eficientes. Um exemplo é a validação automática e periódica da aplicação junto de um serviço detido pelo autor.

Dependendo do grau de integração e funcionalidade final, a utilização de uma aplicação popular também poderá conduzir a pontos extra. Todas estas situações devem ser discutidas com o docente com a devida antecedência.