Relatório de Segurança e Confiabilidade

Trabalho 2

Engenharia Informática

Grupo 51

João David n49448

João Marques n49038

Luís Moreira n49531

Índice

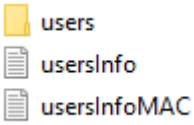
Concretização

UserManager

Programa responsável por criar, actualizar e remover utilizadores, é o único com a capacidade de escrever no ficheiro usersInfo.txt (ficheiro que contém a informação de login dos utilizadores registados) e usersInfoMAC.txt.

Startup

Ao ser iniciado verifica se existe a directoria "(home)\MsgFileG51\server", se não existir assume que é o primeiro boot do programa e cria a directoria com o seguinte conteúdo dentro:

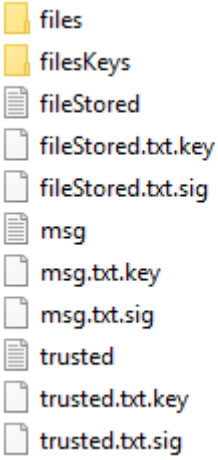


Senão verifica a integridade dos ficheiros usersInfo.txt e usersInfoMAC.txt, se não os encontrar ou estiverem sido alterados o programa lança uma excepção e interrompe a sua execução.

Criação de utilizador

create <username> <password>

Ao ser criado um novo utilizador, (caso não exista ainda um outro utilizador com o mesmo username) é adicionado o seu username:salt:salted_password_hash ao ficheiro usersInfo.txt, é depois criada uma directoria com o seu username dentro da directoria users, com a seguinte estrutura no seu interior:



Actualização de utilizador

update <username> <new_password>

A linha que contem a informação do <username> é alterada de forma a conter um novo salt e novo salted_password_hash

Remoção de utilizador

remove <username>

À linha que contem a informação do <username>, é concatenada a tag ":DEACTIVATED", a linha passa então a ficar no formato "username:salt:salted_password_hash:DEACTIVATED".

De forma a tornar o sistema mais escalável, evitasse estar à procura em todos os locais onde possa aparecer alguma referência a este username, coloca-se apenas a tag no fim da linha, tornado esta operação em O(n), em que n corresponde ao número de utilizadores registados (número de linhas do ficheiro usersInfo.txt).

A conta fica assim desactivada, impossibilitando a criação de uma nova conta com este username, esta implementação acaba por ser também uma medida de segurança, visto que se um novo utilizador se pudesse registar com um username de alguém que já esteve registado no servidor, pode induzir em erro outros utilizadores que já foram amigos do antigo dono do username em dar permissões trusted a este novo utilizador pensando ser o utilizador antigo do username.

Do lado do cliente, este username é como se não tivesse registado, ao fazer download de ficheiros de um username correspondente a um utilizador desactivado mas que antes de ser desactivado tinha dado permissão a outro utilizador para ter acesso aos seus ficheiros, esse utilizador ao tentar fazer download dos ficheiros, é lançado um erro como se o utilizador não fosse trusted. Qualquer operação que envolva o username de um utilizador desactivado vai dar sempre erro.

MsgFileServer

Antes de executar o programa servidor, é obrigatório correr o UserManager, visto que este programa vai criar o ficheiro usersInfo.txt, que vai ser consultado pelo servidor para autenticar os clientes, o servidor apenas tem poder de leitura sobre o ficheiro usersInfo.txt.

O servidor ao ser iniciado, vai verificar a integridade de todos os ficheiros, começa por verificar a integridade de usersInfo.txt com o MAC guardado em usersInfoMAC.txt, se detectar alguma intrusão indevida lança uma excepção e interrompe a execução, caso contrário, procede à leitura deste ficheiro para saber o username de todos os utilizadores activos e coloca esses usernames numa lista, depois para cada username verifica a integridade dos ficheiros de controlo trusted.txt, msg.txt e fileStored.txt, e por fim os ficheiros que o utilizador tenha guardado no servidor, se detectar alguma alteração indevida nestes ficheiros lança uma excepção e interrompe a execução, caso contrário o servidor liga sem problema, e fica à escupa de novas conecções.

MsgFile (cliente)

Execução idêntica à do trabalho 1, os ficheiros transferidos do servidor são armazenados na directoria “(home)\MsgFileG51\client\username”, em que username corresponde ao username do utilizador que se encontra ligado ao servidor e executou o pedido de tranferencia do ficheiro.

Estabelece uma comunicação SSL com o servidor, usando o certificado do servidor importado para a sua truststore.

Segurança

O ficheiro que contém a informação de login dos utilizadores (usersInfo.txt) encontra-se protegido com um MAC, este MAC é posteriormente guardado num ficheiro userInfoMAC.txt, sempre que o programa UserManager pretenda alterar algo no ficheiro, para efeitos de criação, actualização e remoção de utilizador, é previamente verificada a integridade do ficheiro, se for detectada alguma alteração maliciosa, a aplicação lança uma excepção e pára a execução, caso contrário altera o conteúdo do ficheiro e volta a calcular um novo MAC.

No caso do MsgFileServer, vai haver verificação da integridade deste ficheiro ao ligar o servidor, e sempre que um utilizador conectado faça uma operação remota, caso seja detectada uma alteração maliciosa, vai ser lançada uma excepção e abortada a execução.

Sempre que um utilizador envia um ficheiro F para o servidor (independentemente do tipo de ficheiro), este é cifrado com uma chave simétrica K AES gerada aleatoriamente, a chave K é posteriormente cifrada com a chave pública do servidor e armazenada dentro de um ficheiro com o mesmo nome que o ficheiro enviado pelo utilizador com a extensão ".key", o ficheiro que contém a chave não é guardado na mesma pasta que o ficheiro F, visto que essa pasta está reservada para os ficheiros que o utilizador tem acesso, de forma a melhor organizar a estrutura de directorias do servidor, por isso, a chave é guardada na pasta "fileKeys", enquanto que o ficheiro F fica na pasta "files". Existe uma pasta "filesKeys" para cada utilizador, tal como acontece para a pasta "files".

Após o ficheiro ter sido cifrado e armazenado com sucesso, é adicionado ao ficheiro de controlo "fileStored.txt", o nome do ficheiro que acabou de ser guardado pelo utilizador. Desta forma, mesmo que o ficheiro F e o ficheiro que contem a chave que foi usada para cifrar forem apagados, o servidor tem forma de saber que era suposto estar la o ficheiro e lançar excepção referente a violação de integridade dos ficheiro. Sem o "fileStored.txt" a fazer este controlo, um hacker poderia apagar os ficheiros de todos os utilizadores, e o servidor nunca saberia dessa ocorrência.

Outra segurança que esta abordagem oferece é o facto de um eventual intruso com acesso ao sistema de directorias do servidor não conseguir dar acesso a um utilizador B os ficheiros de um utilizador A, ou seja, se não existisse o ficheiro de controlo fileStored.txt, a posse de ficheiros estaria apenas a ser definida pelo que se encontra nas pastas files e filesKeys, desta forma ao copiar o conteúdo destas pastas do utilizador A para as mesmas pastas de outro utilizador B, o utilizador B conseguiria transferir os ficheiros de A. Com a existência do ficheiro de controlo fileStored.txt, um utilizador só consegue transferir os ficheiros que lá se encontram registados, o comando "list" permite saber este mesmo registo.

No entanto, esta implementação continua a ter uma fraqueza, visto que se os utilizadores A e B possuírem ficheiros diferentes com nomes iguais, e como explicado anteriormente, os ficheiros do utilizador A forem copiados para a directoria do utilizador B, o ficheiro de controlo não tem forma de distinguir os dois ficheiros e vai permitir ao utilizador B transferir os ficheiros, desde que tenham o mesmo nome.

Relativamente aos ficheiros de controlo do utilizador (trusted.txt, msg.txt e fileStored.txt), estes são gerados sempre que um utilizador é criado, são assinados (a assinatura é guardada num outro ficheiro com a extensão ".sig"), e são depois cifrados de forma idêntica aos ficheiros enviados pelo utilizador (como explicado anteriormente).

Sempre que uma operação remota evocada por um utilizador necessite de recorrer a estes ficheiros, estes são previamente decifrados e é verificada a assinatura. Caso seja necessário serem alterados, por exemplo na adição de uma nova mensagem, ou utilizador amigo, são novamente assinados e cifrados.

Ao iniciar o servidor, as integridades de todos os ficheiros são verificadas. Nomeadamente os ficheiros dos utilizadores, os seus ficheiros de controlo e o ficheiro que contém a informação de login de todos os utilizadores registados no servidor.

A comunicação entre o servidor e o cliente é efectuada através de um socket SSL, o cliente possui uma truststore com o certificado que foi importado a partir da keystore do servidor que contem a chave privada, desta forma não é possível um atacante fingir ser o servidor.