universidade de aveiro

Segurança

Parte 2

Professores: André Zúquete, andre.zuquete@ua.pt
João Paulo Barraca, jpbarraca@ua.pt

Grupo p1g8
Álvaro Martins 72447
Alvaro.martins@ua.pt

universidade de aveiro

Introdução

Os sistemas de trocas de mensagens são bastante utilizados nos dias de hoje, e são usadas para transmitir informação muitas das vezes pessoal. Ao transmitir informação pessoal tornam se alvos para atacantes que pretendem obter a informação. Como tal a segurança é uma prioridade na implementação de um sistema de trocas de mensagens para impedir que haja fuga de informação pessoal.

Para este projecto o grupo irá implementar um sistema de trocas de mensagens seguro que garante para além da entrega anterior: autenticidade das mensagens trocadas entre clientes , validação do cliente com qual se está a comunicar, controlo de fluxo informação e identificação da máquina de ambos utilizadores.

Secure Instant Messaging System

Conexão ao servidor

O cliente inicia conectando-se ao servidor. O que consiste em estabelecer um canal seguro para troca de mensagens entre cliente servidor

O processo de conexão ocorre em 6 fases. Este processo já foi abordado no último relatório e como tal apenas serão apresentadas as diferenças em relação à primeira entrega, nomeadamente a inserção de certificados, a validação do utilizador e do servidor entre outros.

1. Na primeira fase foi acrescentado o certificado do utilizador e um identificador da máquina no qual o utilizador está a utilizar.
2. Na segunda fase o servidor envia o seu certificado e um desafio gerado aleatoriamente para o cliente assinar. O servidor valida o certificado do cliente e para já considera o cliente como o dono do certificado
3. A partir da terceira fase o utilizador começa a assinar todos os pacotes para o servidor. O cliente valida o certificado do servidor, envia o desafio do servidor assinado e gera um desafio para o servidor assinar.
4. A partir da quarta fase o servidor também assina todos os pacotes enviados e já verifica a assinatura do cliente mas verifica o desafio assinado a ver se o cliente é realmente o dono do certificado que enviou. O servidor nesta fase também assina o desafio que o cliente propôs e envia para o cliente.
5. Na quinta fase o cliente verifica se o servidor é o dono do certificado que enviou
6. Nesta fase não ocorreram alterações

Mensagens do tipo Secure

Todas as mensagens do tipo secure são cifradas à saída, caso sejam destinadas a outro utilizador o conteúdo destinado ao outro cliente também é cifrado. O pacote encapsulado no tipo secure é assinado e é anexado dentro do *payload* do

pacote Secure o HMAC e a assinatura do mesmo para o destinatário. O pacote secure é assinado e anexado o HMAC para o servidor.

Dentro de uma mensagem do tipo secure podem ir os seguintes pacotes:

- List
- Ack
- Client-com
- Client-connect
- Client-disconnect
- Refresh

Será apresentada uma breve descrição do que foi alterado desde a última entrega em cada um dos pacotes.

Listagem de clientes disponíveis

O cliente ao pedir ao servidor uma lista de clientes disponíveis avança para um estado de espera no qual pedidos de listagem posteriores serão ignorados até este se concluir. Para concluir o pedido de list o cliente aguarda um ack por parte do servidor descartando quaisquer list responses que obtenha entretanto. Após o ack do servidor o cliente aguarda a chegada da resposta por parte do servidor. Ao obter a resposta volta para o estado inicial onde qualquer list response ou ack que obtenha é descartado apenas possibilitando uma resposta caso um pedido tenha sido feito impedindo assim injecção de list responses.

A lista de clientes é filtrada de acordo com o modelo Bell-laPadula implementado no sistema.

Conexão com outro utilizador

Para um utilizador se conectar a outro utilizador terá de passar por um processo constituído por 6 fases semelhantes às da conexão com o servidor.

Mensagem de confirmação

É sempre gerada uma mensagem de confirmação após a recepção de um outro tipo de mensagem secure excluindo outras mensagens de confirmação. Como as mensagens de confirmação são do tipo secure e vão assinadas servem para garantir que a mensagem chegou ao destinatário correto (destination validation).

Mensagem de comunicação entre clientes

Estas mensagens contêm um índice que as identifica para o utilizador poder ver a ordem e as poder identificar quando chegar a confirmação que foi recebida pelo destinatário.

Refresh e Mensagem de desconexão

Nada foi alterado no refresh nem no disconnect desde a última entrega

Features

Validação de certificados

Para validar o certificado é validada toda a cadeia de certificados, a delta crl e as crls são obtidas em runtime e é feita a verificação, também é verificada a validade dos certificados.

Autenticação

Para esta entrega o grupo gerou uma CA e depois um certificado para o servidor, assinado pela CA para poder autenticar o servidor. O certificado do utilizador é obtido através do CC.

Os certificados são trocados e validados entre destinatário e remetente no processo de conexão e autenticados com uma variante de “three way handshake” entre destinatário e remetente, ou seja, o destinatário gera um desafio aleatório para o remetente assinar e posteriormente verifica a assinatura e vice versa.

Para autenticar as mensagens, elas vão assinadas pelo remetente e o destinatário verifica a assinatura à chegada com o certificado trocado no processo de conexão.

Validação do destino

A validação do destino é conseguida através das mensagens de confirmação que permitem ao utilizador ver se as mensagens já foram recebidas e pelo destinatário válido

Preservação de identidade

Para preservar identidade é utilizado o cartão de cidadão de um utilizador sendo que cada utilizador é identificado por um número e nome que está no certificado presente no cartão de cidadão. Os certificados são guardados numa base de dados do servidor e sempre que o utilizador se quer conectar são comparados o certificado na base de dados e o certificado disponibilizado pelo utilizador que se quer conectar. Caso o certificado do servidor fique inválido é substituído pelo novo a próxima vez que o utilizador se ligue ao servidor com o novo certificado válido

Controlo de fluxo de mensagens

Para garantir que não há injeção de pacotes, todos os pacotes são “marcados” com um número que é incrementado a cada envio. Este número é comparado com o último recebido no destinatário. Caso apareça um pacote fora de ordem este é descartado.

Controlo de fluxo de informação

Neste sistema foi implementado um modelo Bell-laPadula de 5 níveis (0..4) sendo que quanto maior o nível maior a confidencialidade. Neste projecto o nível é atribuído aleatoriamente para efeitos de demonstração no entanto caso fosse para produção teria de se alterar.

Este modelo é caracterizado por “no write down, no read up”, ou seja, um utilizador de nível superior não poderá comunicar com um utilizador de nível inferior no entanto o contrário já é possível.

Para implementar este modelo filtramos a resposta a um pedido de listagem para apenas enviar os utilizadores de nível semelhante ou maior e apenas possibilitando a conexão caso seja inicia pelo utilizador de nível mais baixo. Caso o utilizador de nível superior tentar comunicar com o de menor nível a mensagem será filtrada no servidor.

Participant consistency

Para esta funcionalidade foi implementado um pequeno módulo que calcula um ID pseudo único do computador e é trocado entre utilizadores no processo de conexão. Ao finalizar é apresentado ao utilizador este ID e o utilizador pode verificar se é um ID desconhecido ou o habitual.

Consistência de conversa

Esta funcionalidade não foi implementada por falta de tempo por parte do grupo. No entanto iria ser implementada com uma base de dados a nível de utilizador semelhante à implementada no servidor.

Conclusão

Ao concluir o projecto foi conseguido implementar um sistema que garante confidencialidade, integridade e autenticidade das mensagens trocadas entre clientes , validação do cliente com qual se está a comunicar e controlo de fluxo informação. Foi possível ao grupo verificar que na entrega anterior o sistema não era seguro contra alguns ataques e vulnerabilidades e implementar mais funcionalidades para impedir algumas dessas vulnerabilidades.

Por falta de tempo o grupo não conseguiu implementar uma base de dados para cada utilizador onde se guardariam as conversas anteriores.

Referências

[1]Segurança em Redes Informáticas, A. Zúquete ISBN: 978-972-722-767-9

[2]https://stackoverflow.com/questions/2461141/get-a-unique-computer-id-in-python-on-windows-and-linux

[3]https://stackoverflow.com/questions/159137/getting-mac-address

NOTA

O projecto apenas foi testado com 2 CC no máximo