Segurança de Redes e Sistemas de Computadores 2015/2016, 2º Semestre

Relatório do Trabalho Prático nº 1

Grupo

Nº 41623 Nuno Neves

Nº 41710 David Gago

Resumo

O trabalho prático desenvolvido, foca-se, numa primeira fase, em manter um sistema de chat seguro para os utilizadores, criando assim canais seguros para a comunicação, encriptando as mensagens enviadas no mesmo de modo a que sejam garantidas propriedades de autenticação, confidencialidade, integridade defendendo da tipologia de ataques com base na referencia OSI X.800. Para garantir estas propriedades foram usados metodos de criptografia simétrica e autenticador de mensagens para garantir a segurança das mensagens que circulam nos canais do sistema. Na segunda fase foi implementado um servidor de autenticação com base no sistema Needham-Shroeder. Caso o utilizador seja autorizado a participar na sessão ele irá receber as credencias sobre um formato de ficheiro pré indicado e só posteriormente poderá aceder a conversa utilizando as credencias fornecidas. As mensagens são verificadas através de um nonce que é enviado pelo cliente e posteriormente recebido pelo servidor.

1. Introdução e contexto do trabalho

Desenvolveu-se um sistema de chat seguro para ser utilizado por utilizadores num sistema de chat em grupo do estilo conferencia. O sistema garante propriedades de autenticação, confidencialidade, integridade e estabelece um ambiente de canais de comunicação fiável e segura. O sistema está implementado sobre transporte TCP (pilha TCP/IP), com extras adicionados para adicionar segurança na comunicação. O trabalho foi realizado com base em material fornecido e foi baseado maioritariamente em programação em JAVA, utilização de sockets TCP/IP e a utilização do suporte JCE (Java Cryptographic Extension). Utilizou-se criptografia simétrica em funções de síntese segura de mensagens e autenticador de mensagens. Para a segunda fase maior parte ja foi descrito. Foi contruido um servidor com base em KDC (Key Distribuition Center) que recebe pedidos de utilizadores para poder participar em sessões. Se o utilizador tiver à disposição deles um ficheiro com o seu username, ser-lhes-á facultado e posteriormente, a partir desse ficheiro, podera usa-lo para participar na sessão.

2. Objetivos da Fase 1

2.1 Objetivos realizados

TABELA A – Requisitos da FASE 1		
Requisito / Objetivo	IMPLEM DO OBJ. S, N ou P	CORREÇÃO DO FUNCIONAMENTO S, N ou P
Proteção das sessões com base na implementação de canais seguros, considerando um modelo de adversário que realiza ataques aos canais de comunicação entre todas as entidades, segundo a tipologia de ataques definida na *framework* X.800 (não considerando ataques á disponibilidade ou ataques do tipo *DoS* ou *DDoS*).	S	
Proteção da confidencialidade e integridade das mensagens nas sessões, mesmo perante possíveis oponentes do tipo "*honest-but-curious system administrators*" que administrem os sistemas onde executam servidores *relay* (*MCServer*) associados a diferentes sessões de *Chat/Conferencing* e possam inspecionar a memória desses processos. Note que se admite que estes oponentes não atentam contra a disponibilidade desses sistemas, nem comprometem a correção do software que executa nos mesmos.	S	

Configuração flexível, através de um ou mais ficheiros de configuração, com indicação dos seguintes parâmetros
SESSIONKEY: Session Key // Representação da Chave Criptográfica
KEYSIZE: size (bytes) // Tamanho da chave
ALG: Algorithm // Algoritmo criptográfico simétrico
MODE: mode // Modo de cifra utilizado
PAD: padding method // Método de *Padding* usado
MAC: Algorithm // Método MAC, HMAC ou CMAC
HMACKEY: HMAC Key // Chave HMAC ou CMAC
HMACKEYSIZE: size (bytes) // Tamanho da chave
	S	

As configurações podem usar qualquer ciphersuite que envolve as configurações anteriores (podendo configurar-se qualquer algoritmo simétrico, em qualquer modo ou *padding* e qualquer MAC (HMAC, CMAC), de acordo com as dimensões válidas de chaves, vetores de inicialização	S	

Na verificação da correção das configurações anteriores, foram testadas combinações envolvendo pelo menos 5 algoritmos criptográficos diferentes com diferentes tamanhos de chave, modo e padding, bem como 3 funções HMAC ou CMAC diferentes	S	
O protocolo de segurança foi implementado corretamente de acordo com a especificação apresentada na aula, para o formato das mensagens e as diversas partes . Se responder não ou parcialmente, deve apresentar depois desta tabela qual o formato da mensagem e das respetivas partes com processamento criptográfico	S	
No anterior protocolo, o campo do AUTENTICADOR não está incluído na parte cifrada da mensagem, sendo este campo adicionado (APPEND) depois da parte cifrada do payload.	S	
No anterior protocolo, o campo do AUTENTICADOR está incluído na parte cifrada da mensagem, sendo assim este campo (APPEND) ao payload plaintext sendo depois todo o payload cifrado	N	
A minha implementação protege os principais nas sessões de possíveis ataques por MESSAGE REPLAYING	S	

2.2 Informação adicional em relação à TABELA A

NÃO EXISTE.

2.3 Proteção contra MESSAGE REPLAYING

Utiliza-se um mapa para associar o último nonce enviado por cada utilizador ao próprio utilizador. Após o envio de cada mensagem, é verificado se o nonce enviado na primeira slot é o correto e após isso é substituído no mapa pelo nonce que se encontra na segunda slot.

2.4 Aspectos valorativos da implementação em relação aos objetivos inicialmente estabelecidos para a FASE 1

NENHUM ASPECTO VALORATIVO A CONSIDERAR.

3 Objetivos da Fase 2

3.1 Objetivos realizados na Fase 2

TABELA B – Requisitos da FASE 2		
Requisito / Objetivo	IMPLEM DO OBJ. S, N ou P	CORREÇÃO DO FUNCIONAMENTO S, N ou P
Procedi à implementação dos objetivos da FASE 2	S	
Na implementação da Fase 2, todo o protocolo de Fase 1 mantem-se igual, com a única exceção de que agora os parâmetros de segurança das sessões são obtidos a partir do SSO Server	S	
Nos objetivos da FASE 2, foi implementado o servidor SSO (SSO Server) de autenticação (que é independente de qualquer servidor *relay* de sessões específicas.	P	
O Servidor SSO (SSO Server) suporta todos os aspectos de configuração ou parametrização indicados, nomeadamente: controlo de acesso às sessões, com parametrização de utilizadores que podem participar numa sessão. Deste modo, os utilizadores para participarem nas sessões devem autenticar-se no servidor de autenticação, que escrutinará da correção da autenticação e decidirá com base na política de permissões e (*enforcement*) com base numa lista de controlo de acesso, se um dado utilizador pode ou não entrar numa dada sessão, podendo nesse caso receber todos os parâmetros associados à suite criptográfica a utilizar.	N	
Se um utilizador for autorizado, receberá dinamicamente as credenciais para participar na sessão (ou seja, toda a informação relativa aos parâmetros criptográficos da sessão e que antes estava configurada em ficheiros com gestão manual) do lado dos clientes	S	

Na informação recebida dinamicamente do servidor SSO (Server SSO), após autenticação e controlo de acesso dos clientes, constam os seguintes parâmetros que são distribuídos aos clientes
 	
SESSIONKEY: Session Key // Representação da Chave Criptográfica
KEYSIZE: size (bytes) // Tamanho da chave
ALG: Algorithm // Algoritmo criptográfico simétrico
MODE: mode // Modo de cifra utilizado
PAD: padding method // Método de *Padding* usado
MAC: Algorithm // Método MAC, HMAC ou CMAC
HMACKEY: HMAC Key // Chave HMAC ou CMAC
HMACKEYSIZE: size (bytes) // Tamanho da chave
	S	

A implementação da FASE 2 suporta de forma correta o protocolo inspirado no modelo de Needham-Schroeder, conforme definido para orientar a implementação	S	
A implementação do anterior protocolo é feita com recurso ao mesmo formato das mensagens que também são usadas no CHAT, apenas variando o tipo de mensagem e o respetivo formato de payload.	S	
Todas as mensagens trocadas entre os clientes e o SSO Server no protocolo de autenticação e estabelecimento dos parâmetros criptográficos da sessão estão protegidas de ataques de Message Replaying	S	
Todas as mensagens trocadas entre os clientes e o SSO Server no protocolo de autenticação e estabelecimento dos parâmetros criptográficos da sessão estão protegidas de forma a garantir-se a integridade do fluxo ordenado de mensagens, usando-se *nonces* e controlos do tipo *challenge/response (u desafio/resposta)*	S	

3.2 Objetivos realizados na Fase 2

"Nos objetivos da FASE 2, foi implementado o servidor SSO (SSO Server) de autenticação (que é independente de qualquer servidor *relay* de sessões específicas)." Não é independente na medida em que o ficheiro de configuração de sessão apenas é criado quando o servidor fica live e comunica com o SSO.

3.3 Proteção contra MESSAGE REPLAYING

Como o envio do SSO para o cliente vai encriptado com a chave que apenas os 2 partilham, mesmo que o atacam replique o pedido efetuado pelo cliente, não e capaz de decifrar a mensagem de resposta

3.4 Proteção de integridade do fluxo de mensagens entre os clientes e o servidor "SSO Server"

A quando do envio da mensagem, o cliente gera um número pseudoaleatório entre 0 e 1000 e envia o respetivo número ao SSO. O SSO soma um a esse número e envia-o de volta. Após o cliente decifrar com a sua chave a mensagem enviada pelo SSO, verifica o nonce de resposta. Se for o mesmo número que enviou mais um, prossegue com a ligação ao sistema.

3.5 Aspectos valorativos da implementação em relação aos objetivos inicialmente estabelecidos para a FASE 2

NENHUM ASPECTO VALORATIVO A CONSIDERAR.