Segurança de Redes e Sistemas de Computadores
2016/2017, 2º Semestre

Trabalho Prático nº 1 (v1.0)

Resumo

Neste trabalho pretende-se desenvolver um sistema de CHAT/MESSAGING seguro para utilização de grupos de utilizadores. O sistema suportará conversas em grupo, baseadas em comunicação segura multiponto, utilizando IP multicast. A segurança da comunicação garantirá aos utilizadores a confidencialidade, integridade e autenticidade das mensagens trocadas, bem como a autenticação e o controlo de acesso às conversas, apenas para utilizadores devidamente autorizados.

Introdução

Pretende-se desenvolver um sistema de CHAT/MESSAGING seguro, suportado em comunicação multiponto por IP *multicast* (IPV4). A noção de "sala de conversa" corresponde a um endereço de grupo *multicast*. As conversas em cada sala correspondem a mensagens trocadas pelo grupo de utilizadores participantes, enviadas para o endereço *multicast* de cada sala. O canal de comunicação multiponto (não orientado à conexão, ou em comunicação em modo *datagrama*) deve assegurar aos utilizadores as seguintes propriedades de segurança, de acordo com definição da *framework* X.800:

- Confidencialidade das mensagens *(connectionless confidentiality)***;**
- Integridade das mensagens (*connectionless integrity);*
- Autenticidade das mensagens enviadas pelos emissores (*data-origin authentication);*

Deste modo, os utilizadores estarão imunes à tipologia de ataques por parte de adversários que que tenham em vista comprometer aquelas propriedades, sejam ataques ativos ou passivos, de acordo com a tipologia de ataques da *framework* X.800. O sistema a implementar garantirá adicionalmente a autenticação dos utilizadores, bem como o controlo de acesso dos mesmos às conversas apenas para utilizadores devidamente autenticados e autorizados a participarem nas respetivas "salas".

O desenvolvimento do sistema será feito a partir da aplicação inicial fornecida (código MCHAT.tgz) sem qualquer das proteções de segurança que se pretendem implementar. Esta aplicação será a base de partida para os desenvolvimentos pretendidos.

Para efeitos da metodologia de desenvolvimento dever-se-á implementar o sistema em duas fases distintas, como a seguir se descreve.

Implementação da FASE 1 (obrigatória): até 14 valores

Nesta primeira fase serão implementadas as propriedades de segurança do canal multiponto. Nesta fase os utilizadores terão ficheiros de configuração instalados manualmente previamente à utilização do sistema, contendo estes:

- A *ciphersuite* a usar nas salas, nomeadamente algoritmos criptográficos simétricos e algoritmos MAC a usar nas salas;
- As chaves criptográficas e demais parâmetros para os algoritmos criptográficos simétricos, de acordo com a *ciphersuite*
- A chave MAC a usar.

Para cada sala em que um utilizador participa, este deve ter a respetiva configuração. Por exemplo, para participar na sala 224.10.10.10, o utilizador terá a informação anterior num ficheiro de configuração com nome 224.10.10.10.crypto. Este ficheiro deverá ter o conteúdo protegido (cifrado) com base num esquema PBEncryption, sendo o conteúdo acessível no início da aplicação, pedindo esta a password do utilizador para este efeito. A parametrização do esquema PBE deve estar noutro ficheiro com nome 224.10.10.10.pbe. Assim, exemplificando, o ficheiro 10.10.10.10.pbe, terá as seguintes linhas (onde # representa um comentário):

```
# Definição do esquema PBE para a sala
PBE:    <algoritmo>             # ex: PBEWithHmacSHA256AndAES_256
SALT:   <valor do salt>         # Valor em hexadecimal, ex: 0x01 0x02 0x03 0x04 ... etc
CTR:    <valor do counter>      # Representado como valor inteiro
```

O ficheiro 10.10.10.10.crypto (cifrado) terá as seguintes linhas

```
# Definição da ciphersuite, chaves e parâmetros
CIPHERSUITE:    <alg/mode/padding>  # ex., AES/CBC/PKCS#5
KEYSIZE:        <valor>         # n. de bits da chave, ex: 256
KEYVALUE:       <valor>         # chave, representada em hexadecimal
MAC:            <mac-alg>       # algoritmo MAC (HMAC ou CMAC), por ex: RC6-GMAC
MACKEYSIZE:     <valor>         # n. de bits da chave do MAC, ex: 256
MACKEYVALUE:    <valor>         # chave MAC, representada em hexadecimal
```

Nota: um aspeto importante na avaliação da FASE 1 é conseguir-se implementar um sistema totalmente parametrizável, podendo os ficheiros utilizarem qualquer *ciphersuite* criptográfica e quaisquer parametrizações apropriadas, desde que estejam corretas para essa mesma *ciphersuite*. Assim, na avaliação do trabalho poderão ser usadas quaisquer combinações crptográficas, desde que corretamente definidas nos ficheiros de configuração. Sugere-se que os alunos testem a implementação com diferentes configurações (que serão reportadas no relatório do trabalho). No critério de avaliação os 12 valores da FASE 1 serão repartidos da seguinte forma: 7 valores pela implementação correta e 1 valor por cada *ciphersuite* diferente que possa ser utilizada em demonstração nas configurações.

A codificação das mensagens no canal deverá ser a seguinte:
`HEADER || PAYLOAD`

HEADER deve conter os seguintes campos (cada um codificado em 8 bits):

`VER || 0x00 || TAMANHO DO PAYLOAD` (sendo VER um número de versão)

PAYLOAD: contem a mensagem cifrada, a respetiva prova de autenticidade e integridade e um *nonce* (gerado pelo emissor como um número aleatório não reutilizável).

Notar que o processamento deve saber distinguir possíveis ataques do tipo *message replaying*, pelo que a codificação do PAYLOAD deverá incluir um NONCE protegido para essa verificação.

Implementação da FASE 2: até 6 valores

Nesta fase será implementado o serviço de autenticação de utilizadores e o controlo de acesso.

Haverá agora um servidor de autenticação e controlo de acesso que autenticará os utilizadores e decidirá, após a autenticação, se o utilizador que se autenticou pode ou não ter acesso à "sala de chat" pretendida. Este servidor pode ser implementado com sockets TCP, JAVA-RMI ou REST, conforme preferência dos alunos, funcionando do seguinte modo.

Quando o utilizador inicia a aplicação (cliente) e após pedir a password ao utilizador, proceder-se-á à autenticação perante o servidor. Para tal o cliente envia ao servidor:

`username || IPMULTICAST || NONCE || H(PWD)`

sendo esta mensagem cifrada com o esquema PBE do ficheiro de configuração com a extensão .pbe (com NONCE um número aleatório calculado pelo cliente).

O servidor deve proceder à autenticação do utilizador, consultando um ficheiro que tem uma tabela com todos os utilizadores na seguinte forma:

user1: pwd1
user2: pwd2
user1: pwd3
...

Neste ficheiro as passwords pw1, pw2, pw3, etc estão sintetizadas a partir das passwords originais dos utilizadores, com base na mesma função H usada pelo cliente na sua mensagem

O servidor poderá assim autenticar os utilizadores. Se a autenticação não estiver correta, será devolvido ao cliente uma mensagem de "autenticação incorreta".

Depois de o fazer deve consultar um outro ficheiro onde está definida a política de controlo de acesso discricionário a cada sala de chat, na forma:

```
224.10.10.10: user1 user2 user3 user4 user 5 ... etc
224.10.10.20: user1 user4 user 5
224.10.10.30: user2 user3 user 5
```

Deste modo o servidor decidirá se o utilizador antes autenticado pode ou não participar na sala de chat que pretende.

Se o utilizador estiver não estiver autorizado recebe uma mensagem referindo não ter permissão para participar nessa sala. Caso seja autorizado receberá toda a informação que na FASE 1 estava no ficheiro de configuração com extensão .crypto, sendo esta informação enviada numa mensagem cifrada com PBEEncryption. O formato desta mensagem é livre mas deve incluir uma resposta ao NONCE previamente enviado pelo cliente na autenticação (por exemplo uma iteração NONCE+1). Ao receber e decifrar esta mensagem, o utilizador pode então entrar na sessão de chat.

Entrega do trabalho

A entrega do trabalho, nas datas já previamente estipuladas, será feita nos moldes que serão indicados oportunamente num ficheiro com instruções a ser publicado no CLIP antes da data de entrega. Para esta entrega os grupos terão:

- Um arquivo TP1-SRSC.tgz com a implementação (código do projeto), com duas subdiretorias separadas, uma com a implementação da FASE 1 e outra com a implementação da FASE 2
- Um relatório, cujo formato (*template*) será publicado no sistema CLIP