**Segurança de Redes e Sistemas de Computadores 2017/2018**

**Ficha de Reporte do Trabalho Prático nº 1 (TP1)**

**Grupo**

|  |  |
| --- | --- |
| **Nº de Aluno** | **Nome (elementos do grupo)** |
| 45617 | André Lopes |
| 45694 | Nelson Coquenim |
| 45020 | Simão Dolores |

1. **Introdução e contexto do trabalho**

Indique X conforme o seu caso

|  |  |  |  |
| --- | --- | --- | --- |
| **Implementação e completude do trabalho** | **SIM** | **NÃO** | **PARC.**  **(Parcialmente)** |
| Foram implementados totalmente todos os requisitos da FASE 1 (ou protocolo STGC/TLP) | x |  |  |
| Foram implementados totalmente todos os requisitos da FASE 2 (ou protocolo STGC-SAP) | x |  |  |
| A minha implementação da FASE 1 (ou implementação do STGC/STGC-TLP) concretiza completamente e exatamente as especificações desse protocolo que constam do enunciado | x |  |  |
| A minha implementação da FASE 2 (ou implementação do STGC/STGC-SAP) concretiza completamente e exatamente as especificações desse protocolo que constam do enunciado |  |  | x |

Se colocou X anteriormente em alguma posição PARC /Parcialmente do quadro, indique porque o fez e porque considera que a implementação é parcial. Se não deixe em branco ou indique N/A

|  |
| --- |
| Não foi implementado o uso do MAC mitigação de ataques contra disponibilidade por esses não constarem no modelo de adversário definido. |

1. **Generalidade do desenvolvimento do protocolo STGC (Subprotocolo STGC-TLP) e sua evidência**

Para suportar a aplicação de teste fornecida (testeMulticast) e para que esta seja protegida pela implementação do protocolo STGC-TLP, dado o código inicial (sem proteção da comunicação) dessa aplicação:

2.1 Apenas foi necessário modificar 1+1 (sender + receiver) linhas de código, em relação ao número de linhas de código da aplicação inicial

2.2 É preciso modificar 1+1 (sender + receiver) linhas de código em relação ao número de linhas de código inicial, tendo ainda que se acrescentar mais 0 linhas de código em relação ao código inicial

Diga em que consiste no essencial a modificação do código da aplicação para ser protegida pela sua implementação com o STGC/TLP:

|  |
| --- |
| Uso de uma nova socket segura.  Sendo necessário acrescentar, para o protocolo STGC-SAP, informação como: endereço do Servidor de Autenticação, o porto do mesmo, identificador do cliente e a sua palavra-passe. |

1. **Caracterização da implementação do protocolo STCG / subprotocolo STGC-TLP**

A minha implementação do subprotocolo STGC foi feita do seguinte modo (caracterize com uma boa síntese, como construiu e desenvolveu o suporte do protocolo STGC/STGC-TLP.

|  |
| --- |
| Criação de uma classe que estende MulticastSocket.  Quando a aplicação pretende transmitir uma mensagem, verificar-se se aquele pacote é suposto ser encriptado ou não, através do campo PAYLOAD\_TYPE presente no Header do pacote:  PAYLOAD\_TYPE = M -> Não encripta  PAYLOAD\_TYPE = S -> Encripta  Consoante o referido anteriormente, configura o pacote (encriptando ou não) e depois executa a função de send da Superclasse MulticastSocket.  No caso de encriptar, esta é o formato do PAYLOAD:  E (KS, [ Mp || MACKM (Mp) ] )  Mp = [id || nonce || M]  Ks: chave de sessão (sessão de grupo multicast segura STGC)  KM: Chave de autenticidade e integridade na função MAC |

1. **Comprovação da correção da implementação do protocolo STGC-TLP**

|  |  |  |
| --- | --- | --- |
| **4.1 Utilizei como aplicação de comprovação e prova do funcionamento da minha implementação STGC/STGC-TLP** | **SIM** | **NÃO** |
| a) a aplicação MCHAT | x |  |
| b) a aplicação STREAMING |  | x |

|  |  |  |
| --- | --- | --- |
| **4.2 Nas minhas observações experimentais, a aplicação protegida pela minha implementação do protocolo STGC/STGC-TLP:** | **SIM** | **NÃO** |
| a) Funciona corretamente | x |  |
| b) Funciona bem mas apenas parcialmente |  | x |

**Justifique, apenas no caso de ter respondido SIM a 4.2 b). Se não deixe em branco ou coloque N/A**

|  |
| --- |
|  |

1. **Flexibilidade e configuração de parametrizações de segurança para a execução do protocolo STGC/STGC-TLP**

A minha implementação STCG/STGC-TLP segue as especificações do enunciado do trabalho, sendo *os endpoints de comunicação* parametrizáveis pelos seguintes ficheiros (configuração):

|  |  |  |
| --- | --- | --- |
| **5.1 Ficheiro de configuração ciphersuite.conf** | **SIM** | **NÃO** |
| **5.2 keystore.jceks** | **x** |  |

**5.3** Uma configuração tipo no ficheiro ciphersuite,conf pode ser estabelecida do seguinte modo (exemplifique):

|  |
| --- |
| <?xml version="1.0"?>  <groups>  <group ip="224.10.10.10">  <mackProvider>BC</mackProvider>  <mackCipher>DES</mackCipher>  <cipherConfig>AES/CBC/PKCS5Padding</cipherConfig>  <cipherProvider>SunJCE</cipherProvider>  <keySize>192</keySize>  <keyValue>\*</keyValue>  <mackKeySize>64</mackKeySize>  <mackKeyValue>\*</mackKeyValue>  </group>  <group ip="224.20.20.20">  <mackProvider>BC</mackProvider>  <mackCipher>DES</mackCipher>  <cipherConfig>DES/CBC/PKCS7Padding</cipherConfig>  <cipherProvider>BC</cipherProvider>  <keySize>64</keySize>  <keyValue>\*</keyValue>  <mackKeySize>64</mackKeySize>  <mackKeyValue>\*</mackKeyValue>  </group>  </groups> |

**5.4** Com o suporte de configuração **ciphersuite.conf** e com a geração / utilização adequadas (correspondentes) do **keystore.jceks**, verifiquei que se suportarão de forma flexível quaisquer combinações criptográficas. No meu caso testei e comprovei experimentalmente as seguintes:

LISTA DE CIPHERSUITES testadas com sucesso: (ALG/MODO/PADDING):

Todas estas combinações

(DES,AES,DESede,RC6,Blowfish,Twofish,Camellia,RC2)/(ECB,CBC,CFB,CTR)/(PKCS5Padding, NoPadding)

LISTA DE MACs (HMACs ou CMACs) testadas com sucesso:

HMACS: HMacSHA1, HMAC/SHA384, HMAC-SHA3-224, HMAC-SHA3-256, HMAC-SHA512

CMACS: DES

1. **RESPONDA A ESTA SECÇÃO APENAS SE IMPLEMENTOU O SUB-PROTOCOLO STGC-SAP, de acordo com os requisitos do enunciado. Se não, passe ao ponto 7 (Conclusões)**

**6.1 Apresente (usando notação apropriada) a especificação (o mais completa possível) das mensagens trocadas no contexto do processamento do subprotocolo STGC/SAP:**

|  |
| --- |
| **Ronda 1: Client > AS: Formato da mensagem com os componentes criptográficos e sua descrição:**  Cliente ID || NonceC || IPMC || AutenticadorC.length || **AutenticadorC**  **AutenticadorC:**  PBE[ **Kpbe**, Cliente ID || NonceC || IPMC || SHA-512(pwd) || MACk(**X**)]  **Kpbe** = SHA-512(pwd)  **X** = Nonce C || IPMC || SHA-512(pwd)  **k (mac) =** MD5 [NonceC || SHA-512(pwd)] |
| **Ronda 12 AS > Client: Formato da mensagem com os componentes criptográficos e sua descrição:**  **E[*KPBE*, ( NonceC+1 || NonceS || TicketAS ) || MACK (X) ]**  ***KPBE* =** Hpwd || NonceC+1 |

**6.2 O servidor AS possui configurações com os seguintes ficheiros, conforme a especificação do enunciado:**

|  |  |  |
| --- | --- | --- |
| **Ficheiro de configuração** | **SIM** | **NÃO** |
| **ciphersuite.conf**  **//gestão de ciphersuites utilizáveis para as sessões** | **x** |  |
| **keystore.jceks**  **//chaves (criptográficas simétricas ou para MACs – HMACs ou CMACs)** | **x** |  |
| **users.conf**  **//Utilizadores registados que podem participar em grupos multicast seguros STGC** | **x** |  |
| **dacl.conf**  **//configuração de listas de controlo de acesso (DAC) de utilizadores que podem participar em cada grupo multicast definido como grupo seguro STGC** | **x** |  |
| **stgcsap.conf**  **//configuração criptográfica para possíveis construções PBEncryption e MACs para o protocolo STGC-SAP** | **x** |  |

**6.3 A minha implementação do protocolo STGC-SAP pode ser configurável no ficheiro stgcsap.conf, tendo sido verificado experimentalmente com configurações envolvendo:**

|  |  |  |
| --- | --- | --- |
| **PBE (Password-Based Encryption)** | **SIM** | **NÃO** |
| PBEWithSHAAnd3KeyTripleDES | **x** |  |
| BEWITHSHA256AND256BITAES-CBC-BC | **x** |  |
| PBEWITHSHA-1AND256BITAES-CBC-BC | **x** |  |
| PBEWithHmacSHA224AndAES\_256 //não existe no provider BC |  |  |
|  |  |  |
| **MACS (HMACS)** | **SIM** | **NÃO** |
| HMacSHA1 | **x** |  |
| HMAC/SHA384 | **x** |  |
| HMAC-SHA3-224 | **x** |  |
| HMAC-SHA3-256 | **x** |  |
| HMAC-SHA512 | **x** |  |
| **MACS (CMACS)** | **SIM** | **NÃO** |
| SKIPJACKMAC |  | **x** |
| AESGMAC |  | **x** |
| RC6GMAC |  | **x** |
| RC5MA |  | **x** |
| DES | **x** |  |

**6.4 Indique em que consiste o formato de um TocketAS (devolvido na ronda 2 do subprotocolo STGC-SAP). Pode copiar a estrutura de dados que o descreve:**

private Key ks; //symmetric encryption key

private Key km; //key for MAC

private byte[] ivSpec;

private String cipherProvider;

private String cipherSuite;

private String macCipher;

private String macProvider;

private String cipherMode;

**Conclusões e aspectos complementares**

**Inclua as conclusões sobre o seu desenvolvimento do TP1, podendo realçar aspectos complementares ou diferenciados da sua implementação. Se achar relevante pode argumentar sobre aspectos qualitativos que considera valorizáveis**

* 1. **Conclusões resumidas:**

Durante a implementação deste protocolo (STGC) surgiram diversas dificuldades. A verdade é que as dificuldades advieram, principalmente, da implementação do subprotocolo STGC-SAP já que este apresenta uma complexidade muito maior, nomeadamente nos constituintes das mensagens encriptadas trocadas entre o cliente e o servidor de autenticação.

No final, a solução apresentada permite cobrir o modelo de adversário definido para o protocolo STGC, garantindo, para esse efeito, todas as propriedades de segurança necessárias.

* 1. **Aspectos complementares a salientar:**

-O código, por vezes, pode parecer algo confuso na encriptação devido às diferentes partes da mensagem encriptada e à necessidade de utilizar streams para construir as mensagens;

-Não foi usado o mail para o ClientID, mas apenas o username.

* 1. **Argumentação sobre fatores diferenciados e qualitativos implementados no TP1**

A modularidade da solução é uma das vantagens da solução, sendo que existem ficheiros de configuração, de forma a evitar ciphersuites (entre outros…) *hardcoded*, e a modulação das classes e funções está de acordo com as melhores práticas da programação orientada a objectos.

Por outro lado, é extensível, sendo esta característica comprovada pelas prova de conceito, efetuadas utilizando o MCHAT.