Um Mecanismo Para Distribuição Segura de Vídeo MPEG

CÍNTIA BORGES MARGI
GRAÇA BRESSAN
WILSON V. RUGGIERO

EPUSP - Escola Politécnica da Universidade de São Paulo
LARC – Laboratório de Arquitetura e Redes de Computadores
CEP: 05508-900 São Paulo (SP)
{cbmargi, gbressan, wilson}@larc.usp.br

Resumo: Este trabalho faz um levantamento dos principais aspectos envolvidos na distribuição segura de material multimídia e propõe um mecanismo de distribuição e reprodução de vídeos MPEG que atenda a estes requisitos. Diversos métodos de criptografia para vídeos MPEG são analisados e comparados. A partir dos principais requisitos, um mecanismo para distribuição segura de vídeo MPEG é proposto, inclusive com a especificação de um servidor e de um visualizador.

Palavras-Chave: distribuição, MPEG, segurança, confidencialidade, criptografia, *hash*

1 Introdução

Com o aumento da distribuição de material multimídia através da Internet, principalmente com conteúdos valiosos como é o caso do ensino a distância, a discussão dos aspectos de segurança envolvidos torna-se um assunto muito importante. Esta questão fica mais interessante quando se leva em conta a distribuição de materiais multimídia com acesso controlado em um ambiente de decodificação em tempo real.

Utilizar material multimídia na Web significa integrar e disponibilizar vídeos, áudio, textos, imagens e/ou animações. Cada uma destas mídias possui características diferentes tanto na sua codificação, como no modo de distribuição. Assim, faz-se necessário analisar estes aspectos para cada uma destas mídias isoladamente e os seus conseqüentes aspectos de integração.

Discutir os aspectos de segurança envolvidos na distribuição de material multimídia significa considerar algumas questões principais como: controle de acesso, integridade e sigilo. Estas questões estão interligadas, já que o sigilo torna-se relevante quando o acesso ao material é controlado, ou seja, somente usuários autorizados podem utilizá-lo.

Os textos, animações, desenhos e simulações podem ser transmitidos com segurança através de SSL (*Secure Sockets Layer*), utilizando criptografia e certificados digitais. O uso de certificados digitais garante a autenticidade do servidor, e o uso de criptografia garante a confidencialidade e a integridade das informações. Portanto, esta questão possui uma solução bastante satisfatória.

Este trabalho propõe-se a analisar os aspectos de segurança envolvidos, levantar os requisitos e especificar um mecanismo seguro para distribuição e reprodução de material multimídia, mais especificamente de vídeo MPEG.

2 Segurança na Distribuição de Material Multimídia

Os aspectos de segurança a serem enfocados neste trabalho envolvem sigilo, controle de acesso e integridade.

Os serviços de segurança caracterizam os diferentes aspectos de um sistema de computadores [Stallings (1998)], tais como:

- **Autenticidade:** Requer que a origem ou o originador de uma mensagem seja corretamente identificado. A verificação da autenticidade é necessária após todo processo de identificação, seja de um usuário para um sistema, de um sistema para o usuário ou de um sistema para outro sistema. A troca de certificados digitais permite garantir a autenticidade do servidor e do cliente envolvidos na transação.
- **Integridade:** Consiste em proteger a informação contra modificação sem a permissão explicita do

proprietário daquela informação. A modificação inclui ações como escrita, alteração de conteúdo, alteração de *status*, remoção, criação e o atraso de informações transmitidas. Deve-se considerar a proteção da informação nas suas mais variadas formas: armazenada em discos, fitas de *backup*, etc... A integridade pode ser verificada através do *hash* da mensagem. As funções de *hash* são unidirecionais e possuem saída de tamanho fixo.

- **Confidencialidade:** Consiste em proteger a informação contra leitura ou cópia por alguém que não tenha sido explicitamente autorizado pelo proprietário daquela informação. A informação deve ser protegida qualquer que seja a mídia que a contenha: impressa, digital, etc... Este tipo de segurança inclui não apenas a proteção da informação como um todo, mas também de partes da informação que podem ser utilizadas para inferir sobre o todo. O sigilo, ou confidencialidade, é obtido através de criptografia.
- **Controle de Acesso**: Consiste na capacidade de se permitir ou negar acesso aos serviços e recursos oferecidos pelo sistema. Acessos desconhecidos ou feitos por pessoas não autorizadas podem significar a necessidade de uma verificação de todos os recursos envolvidos em busca de possíveis estragos que possam ter sido causados ao sistema, mesmo que aparentemente nada tenha ocorrido. O controle de acesso pode ser implementado através de validação de senhas, ou através de identificação por certificados digitais.

3 O Padrão MPEG

Como o volume de dados de vídeo é muito grande (1 segundo de vídeo na resolução de 640 x 480 resulta em 27 MB), tornam-se necessárias técnicas de compressão. O padrão MPEG-1, criado em 1991, foi desenvolvido para armazenar sinais digitais de áudio e vídeo colorido com qualidade VCR (Vídeo Cassete Record), e ser transmitido a uma taxa de 1,5 Mbps [LeGall (1991)]. O padrão MPEG trata separadamente vídeo e áudio, especificando como estes sinais são associados e sincronizados, possuindo assim três níveis: a camada de sistema, a camada de vídeo e a camada de áudio.

A compressão de vídeo consiste em eliminar as informações redundantes (correlatas). Estas correlações podem aparecer de duas formas: correlação espacial e correlação temporal. A correlação espacial é observada em uma mesma imagem, ou seja, são as informações redundantes que aparecem em uma imagem, como por exemplo a cor de fundo de uma imagem. Para eliminar a correlação espacial utiliza-se a Transformada Discreta de Cosseno (DCT), seguida da quantização dos coeficientes obtidos. Já a correlação temporal é observada em dois quadros consecutivos; por exemplo a primeira cena mostra uma sala com móveis e uma pessoa, enquanto na segunda cena aparece a mesma sala, porém a pessoa mudou de lugar. Para eliminar a correlação temporal, utiliza-se o processo chamado de Compensação de Movimento, que é o emprego da técnica DPCM (*Differential Pulse Code Modulation*), codificando apenas as diferenças encontradas entre os quadros.

Em MPEG-1 a imagem é dividida em blocos 16 x 16 amostras para luminância, e blocos de 8 x 8 amostras para cada sinal de crominância. Um macrobloco é composto por um bloco de luminância (4 x (8 x 8) amostras) e dois blocos de crominância (1x (8 x 8) + 1x (8 x 8) amostras), conforme observa-se na Figura 1. O vetor de movimento indica a translação espacial de um bloco para o outro, sendo utilizado na Compensação de Movimento para eliminar a correlação temporal.

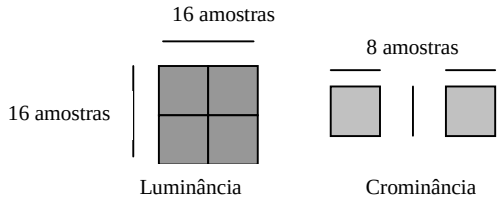

Figura 1: Constituição do Macrobloco MPEG

As cadeias de vídeo podem ter três tipos de quadros:

- quadro I (*intra-frame*): é um quadro codificado somente com informações da imagem, não dependendo de qualquer quadro passado ou futuro;
- quadro P (*forward predicted frame*): este quadro é codificado relativamente ao quadro de referência precedente mais próximo (quadro I ou quadro P);
- quadro B (*bi-directional predicted frame*): sua codificação é feita relativa ao quadro de referência precedente mais próximo (quadros I ou P), ou ao quadro de referência sucessivo mais próximo, ou a ambos.

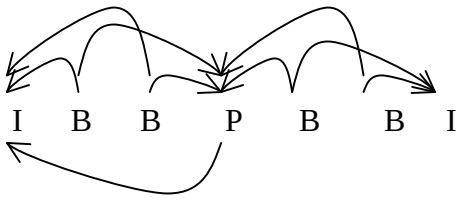

Figura 2: Interdependência de Quadros para uma Seqüência MPEG

Uma seqüência típica de quadros MPEG é apresentada na Figura 2, onde a dependência entre os quadros I, P e B pode ser observada [Mitchel *et al.* (1996)]. Note que se um quadro I não é decodificado corretamente, todos os quadros seguintes apresentarão erros, até a decodificação do próximo quadro I.

A camada de vídeo MPEG é dividida em seis camadas: Camada de Seqüência de Vídeo, Camada de Grupos de Imagens (GOP), Camada de Imagem, Camada de *slice*, Camada de Macroblocos e Camada de Blocos, conforme observa-se na Figura 3.

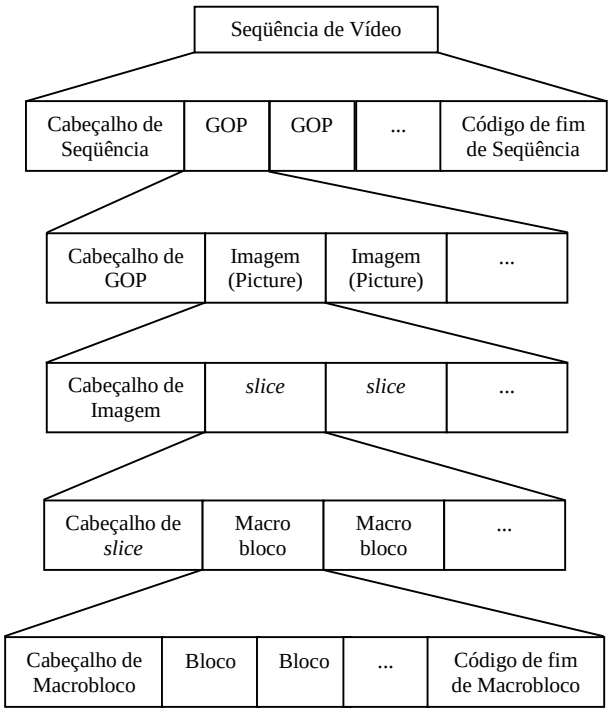

Figura 3: Estrutura da Camada de Vídeo MPEG

Cada uma destas camadas é identificada pelo seu cabeçalho, cujos valores podem ser observados na Tabela 1 [Mitchel *et al.* (1996)].

Tabela 1: Códigos de início de vídeos MPEG

Nome do Código de Início	Valor em Hexadecimal
extension_start code	000001B5
group_start_code	000001B8
picture_start_code	00000100
Reservado	000001B0
Reservado	000001B1
Reservado	000001B6
sequence_end_code	000001B7
sequence_error_code	000001B4
sequence_header_code	000001B3
slice_start_code 1	00000101
...	...
slice_start_code 175	000001AF
user_data_start_code	000001B2

O processo de compressão MPEG segue os seguintes passos:

- processo de identificação dos quadros;
- preparação dos blocos de dados;
- codificação: transformada discreta de coseno (DTC), quantização, supressão de seqüências repetidas (aplicada em zig-zag) e codificação de Huffman.

4 Criptografia para MPEG

Existem diversos mecanismos de criptografia para MPEG, sendo cada um deles com enfoques diferentes. Dentre estes mecanismo podem ser citados: Criptografia Simples, Criptografia Seletiva, Algoritmo de Permutação Zig-zag e VEA (*Video Encryption Algorithm*).

4.1 Criptografia Pura ou Simples

No mecanismo de criptografia pura (*Naive Encryption*), os vídeos MPEG são tratados como dados, ou seja, não são consideradas as características da codificação MPEG. O arquivo MPEG é criptografado utilizando um algoritmo de criptografia convencional, como o DES ou IDEA, e, então, o arquivo MPEG é enviado. Após receber o arquivo mpeg, este é decriptografado e o arquivo obtido pode ser assistido. Observe que o arquivo mpeg estará desprotegido (decriptografado) no disco do usuário.

Este mecanismo proporciona um nível de segurança alto, já que a dificuldade em quebrar o algoritmo é aquela apresentada ao tentar quebrar o DES ou o IDEA. Outra característica deste mecanismo é não alterar o tamanho do arquivo após a criptografia, mas é muito lento [Qiao *et al.* (1998)], além de ocorrer aumento no atraso durante a decodificação da cadeia de vídeo [Spannos *et al.* (1996)], tornando inviável utilizá-lo em aplicações de tempo real.

4.2 Criptografia Seletiva

A criptografia seletiva procura utilizar as características das cadeias de vídeo MPEG para diminuir a quantidade de informações criptografadas. Os quadros I são aqueles que carregam mais informações, enquanto os quadros P e B representam variações da imagem em quadros I adjacentes. Assim, se os quadros I forem criptografados será difícil compreender o conteúdo de uma cadeia de vídeo. Alguns mecanismos também permitem criptografar os quadros I e P, ou todos os quadros (I, P e B).

Porém, quando criptografa-se somente os quadros I e executa-se o arquivo em *player* MPEG convencional

que suporte erros, ainda é possível perceber o conteúdo do vídeo [Agi *et al.* (1996)]. Uma solução proposta [Agi *et al.* (1996)] é aumentar a freqüência dos quadros I, mas isto diminui a compressão, aumentando o tamanho do arquivo.

Dentre os diversos trabalhos que implementam este tipo de mecanismo podem ser citados: SE_MPEG, Aegis e SECMPEG.

SE_MPEG [Li *et al.* (1996)]: O mecanismo MPEG seguro proposto implementa proteção através de um esquema de criptografia baseado nas seqüências de codificação MPEG. O esquema de criptografia é baseado no PGP, ou seja, é feita criptografia simétrica (algoritmo IDEA) dos quadros e a chave é distribuída através de um esquema de criptografia assimétrica (algoritmo RSA). A criptografia pode ser feita somente no quadros I, nos quadros I e P, ou em todos os quadros I, P e B. Um aumento na proteção implica em maior overhead na criptografia, pois aumentam os quadros a serem criptografados. A encriptação do vídeo MPEG é feita criptografando cada um dos quadros, e mantendo as seqüências de início e fim de código do arquivo MPEG. Quando escolhe-se entre criptografar somente quadros I, quadros I e P, ou os quadros I, P e B, obtém-se uma hierarquia de proteção.

Para criptografia somente de quadros I, a degradação da performance (fps) devido à criptografia varia de 10 a 15% [Li *et al.* (1996)]. Já para os quadros I, P e B, varia de 13 a 22% [Li *et al.* (1996)]. Apesar de parecerem índices altos de degradação, segundo [Li *et al.* (1996)], estes resultados são compatíveis com aplicações para Internet.

Aegis [Spannos *et al.* (1995)] [Spannos *et al.* (1996)]: No esquema de criptografia proposto somente os quadros I de todos os grupos de quadros em uma cadeia de vídeo MPEG são encriptados, utilizando o algoritmo DES. O Aegis também encripta o cabeçalho de seqüência de vídeo. O cabeçalho de seqüência de vídeo contém os parâmetros de inicialização da decodificação, como altura e largura do quadro, taxa de quadros, taxa de bits e tamanho do *buffer*. Encriptar o cabeçalho dissimula a identidade de uma cadeia MPEG, fazendo com que esta torne-se irreconhecível. O código de seqüência final também é encriptado no Aegis, dificultando ainda mais o reconhecimento de uma cadeia MPEG. Devido a estas diferenças, um *player* MPEG convencional não é capaz de decodificar corretamente uma seqüência de vídeo codificada pelo Aegis, sendo que as imagens de fundo aparecem borradas e sem nitidez [Spannos *et al.* (1996)].

Segundo uma simulação do mecanismo feita no trabalho, o desempenho do Aegis é bastante próximo ao de um sistema sem criptografia, já que este é capaz de manter constante o atraso devido à criptografia. Um sistema com criptografia completa não mantém o atraso constante, pois demora para processar as informações de entrada, acumulando atrasos. Apesar de os atrasos obtidos com Aegis serem muito próximos daqueles obtidos com um *player* convencional, o nível de segurança é aceitável, mas não é adequado para aplicações sensíveis [Agi *et al.* (1996)], já que com *players* com suporte a erros ainda é possível identificar a imagem criptografada.

SECMPEG [Meyer *et al.* (1995)]: Este projeto propõe uma variação do padrão MPEG para a transmissão segura de vídeo, que incorpora criptografia seletiva e informações adicionais no cabeçalho. SECMPEG pode utilizar os algoritmos DES e RSA para a criptografia, e faz um cálculo de CRC para verificar a integridade do conteúdo. Implementa quatro níveis de segurança:

- 1º nível: encripta todos os cabeçalhos;
- 2º nível: encripta todos os cabeçalhos mais os coeficientes DC e os termos AC dos blocos I;
- 3º nível: encripta os quadros I e os blocos I dos quadros P e B;
- 4º nível: encripta todos os campos.

4.3 Algoritmo de Permutação Zig-Zag

O mecanismo proposto associa a criptografia a compressão da imagem e do vídeo (JPEG e MPEG) [Tang (1996)]. Este mecanismo de criptografia utiliza uma lista randômica de permutação para fazer o mapeamento dos blocos 8 x 8 no vetor 1 x 64, ao invés de fazê-lo em zig-zag (que é utilizado pelo padrão MPEG).

A partir de quatro experimentos com a ordem dos coeficientes DC e AC no vetor 1 x 64 concluiu-se que: a posição do coeficiente DC é importante; a imagem ainda é compreensível se o coeficiente DC for zero e os coeficientes AC forem permutados em zig-zag; o último coeficiente AC pode ser mudado para zero através da matriz de quantização sem prejuízo à qualidade da imagem.

Outro mecanismo estudado é a divisão do coeficiente DC ($d_0d_1d_2d_3d_4d_5d_6d_7$) em duas partes com quatro bits, sendo uma delas colocada no lugar do coeficiente DC ($d_0d_1d_2d_3$) e outra no lugar do último coeficiente AC ($d_4d_5d_6d_7$). Assim, a codificação / criptografia utiliza este mecanismo, e em seguida aplica a lista de

permutação ao vetor 1 x 64, ao invés da permutação em zig-zag.

Este algoritmo aumenta consideravelmente o tamanho das cadeias de vídeo, já que, quando altera-se a ordem do vetor 1x64, perde-se capacidade de compressão (esta é maximizada quando aplica-se a lista de permutação em zig-zag, o que aumenta o número de símbolos repetidos de Huffman).

Este mecanismo de criptografia é vulnerável ao tipo de ataque de texto limpo conhecido. Por este motivo são realizadas algumas modificações: aplica-se uma função de criptografia ao coeficiente DC, e são geradas duas listas de permutação, que são aplicadas segundo um sorteio (*flip coin*).

A Tabela 2 mostra o desempenho do algoritmo para dois vídeos: flower.mpg e tennis.mpg.

Tabela 2: Desempenho do Algoritmo de Permutação Zig-Zag

Tempo para codificação / Vídeo	Algoritmo Original (sem criptografia)	Algoritmo de Permutação Zig-Zag
flower.mpg	37.985 seg	37.969 seg
tennis.mpg	14.213 seg	14.403 seg

4.4 VEA (Video Encryption Algorithm*)*

O algoritmo *Video Encryption Algorithm* (VEA) utiliza o comportamento estatístico do vídeo comprimido [Qiao *et al.* (1997)]. A análise estatística feita com as cadeias de vídeo MPEG trata as cadeias de vídeo como *bytes*. A primeira observação feita é que a freqüência de ocorrência dos valores destes *bytes* (0 a 255) é praticamente a mesma para qualquer valor do *byte*. Analisando esta distribuição para meio *byte*, em qualquer posição da cadeia, não ocorre nenhuma alteração na distribuição de freqüência. Ainda, observa-se que diferentes cadeias MPEG possuem o mesmo comportamento.

Outro estudo realizado é relacionado a freqüência de ocorrência de diagramas (pares de números adjacentes). Esta análise divide o quadro I em porções, e então verifica-se o número de ocorrências do par de maior freqüência na porção. Se um destes pares se repetir, então um diagrama se repetiu. Observou-se que não há nenhum padrão de *byte* repetido com porções de 1/16 de um quadro I. Esta informação é relevante para o desenvolvimento do algoritmo VEA.

O algoritmo VEA assume que uma porção do quadro I terá a seguinte forma: $a_1a_2...a_{2n-1}a_{2n}$. Separa-se os *bytes* pares dos *bytes* ímpares, obtendo duas novas cadeias (lista par e lista ímpar). Aplica-se a função Ou-exclusivo entre as listas par e ímpar, obtendo-se $c_1c_2...c_n$. Escolhe uma função de criptografia E, e aplica-se a lista par. O texto criptografado é $c_1c_2...c_n$ E ($a_2a_4...a_{2n}$).

Neste trabalho uma comparação entre o Aegis e o VEA é feita, sendo que o VEA proporciona um ganho de 47% no tempo total de criptografia em relação a criptografia com o IDEA [Qiao *et al.* (1997)].

4.5 Permutação Pura

Os resultados estatísticos que permitiram o desenvolvimento do VEA, também validam o uso da Permutação Simples. A permutação simples embaralha os *bytes* das cadeias por permutação. A cardinalidade da chave de permutação depende do nível de segurança desejado, podendo variar de 64 números até 1/8 de um quadro I [Qiao *et al.* (1998)].

4.6 Comparação Entre Os Mecanismos Descritos

Alguns dos mecanismos de criptografia descritos podem alterar o tamanho da cadeia de vídeo MPEG, como o de Permutação em Zig-Zag. O nível de segurança de cada um dos mecanismos é diferente, além do tempo necessário para a criptografia [Qiao *et al.* (1998)] (ou velocidade de criptografia).

Assim, pode-se comparar estes algoritmos segundo três parâmetros: velocidade de criptografia, nível de segurança e tamanho das cadeias de vídeo. A Tabela 3 mostra os resultados desta comparação [Qiao *et al.* (1998)], parecendo o VEA ser o melhor algoritmo.

Tabela 3: Comparação dos Algoritmos de Criptografia MPEG

Algoritmo	Nível de Segurança	Velocidade	Tamanho das Cadeias
Criptografia Pura	Alto	Lento	Sem alterações
Criptografia Seletiva	Moderado	Rápido	Aumenta
Permutação Zig-zag	Muito baixo	Muito rápido	Aumenta muito
VEA	Alto	Rápido	Sem alterações
Permutação Pura	Baixo	Super rápido	Sem alterações

5 Um Mecanismo Seguro para a Distribuição de Material Multimídia

Uma vez analisados os mecanismos de criptografia existentes, é possível levantar quais as características importantes para um mecanismo seguro de distribuição de vídeo MPEG. A implementação deste mecanismo resultará em: um *player* MPEG seguro, o **S/Viewer**, na definição do esquema de codificação / criptografia

MPEG para o servidor de vídeo, o **S/Server**, e do protocolo de acesso ao vídeo.

5.1 Requisitos

Este mecanismo de distribuição segura de vídeo MPEG deve considerar os seguintes aspectos de segurança:

- Controle de Acesso;
- Reprodução de Material Autenticado;
- Diversos Níveis de Segurança.

5.1.1 Controle de acesso

Os vídeos armazenados no servidor são divididos em dois grupos: vídeos de acesso público e de acesso restrito. Os vídeos de acesso público podem ser reproduzidos por qualquer usuário, fazendo parte dos privilégios de todos os usuários que acessem o servidor de vídeo após a sua identificação. Já os vídeos de acesso restrito, como é o caso daqueles pertencentes aos cursos *online*, possuem associados a eles uma lista com os usuários que tem a autorização para reproduzi-los. A lista de permissão de acesso de um vídeo é criada quando da inserção do mesmo no servidor de vídeo, podendo ser alterada por seu responsável posteriormente.

A identificação do usuário é feita através de seu certificado digital, e os seus privilégios são determinados checando as listas de acesso dos vídeos.

5.1.2 Reprodução de Material Autenticado

Para iniciar a reprodução de um vídeo MPEG seguro, o S/Viewer deverá solicitar o certificado digital do servidor de vídeo, garantindo desta forma a autenticidade do material a ser reproduzido.

Além disto, o *player* também deve verificar a integridade do vídeo a ser reproduzido. A integridade do vídeo pode ser checada através da verificação do *hash* do mesmo, que está encriptado (assinatura digital).

5.1.3 Níveis de Segurança

Como existem diferentes requisitos de segurança para os vídeos disponíveis no servidor, é necessário definir diversos níveis de segurança. Estes podem ser obtidos através de dois modos:

- utilizando diferentes esquemas de codificação / criptografia MPEG;
- através do uso de diferentes chaves de criptografia para diferentes clientes.

Conforme descrito na seção 4, os diversos esquemas de criptografia MPEG possuem diferentes níveis de segurança e diferentes velocidades de criptografia, conforme observa-se na Tabela 3 [Qiao et al. (1998[b])]. Assim, variando os algoritmos de criptografia é possível obter diferentes níveis de segurança para que haja um compromisso adequado com a velocidade de decodificação.

Para um dado esquema de criptografia MPEG escolhido, ainda é possível utilizar diferentes chaves de criptografia. Assim, cada cliente pode reproduzir o vídeo MPEG seguro solicitado utilizando uma chave diferente, ao invés de todos os clientes utilizarem a mesma chave para o mesmo vídeo. Este modo proporciona um bom nível de segurança.

5.2 Especificação

A especificação do mecanismo de distribuição segura de vídeo MPEG consta de três partes: do protocolo de acesso, do servidor de vídeo (**S/Server**) e do *player* (**S/Viewer**). As próximas seções tratam desta especificação.

5.2.1 Protocolo de Acesso

Para acessar um vídeo disponível no **S/Server** (servidor de vídeo), o **S/Viewer** (o *player* MPEG seguro proposto) irá estabelecer uma conexão com o mesmo. A Figura 4 ilustra as mensagens trocadas durante o processo de acesso ao vídeo.

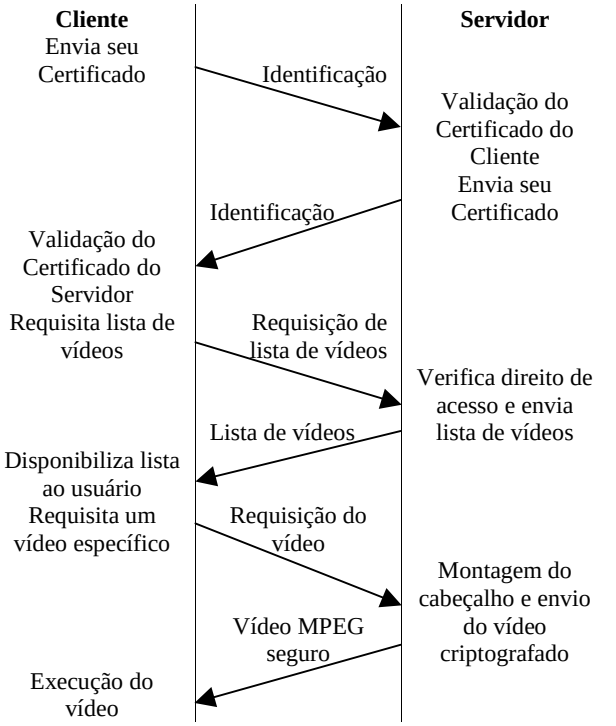

Figura 4: Diagrama de Tempo das Mensagens Trocadas entre o Cliente e o Servidor para Acesso ao Vídeo MPEG Seguro

A primeira mensagem que o cliente envia contém um campo de controle, o seu certificado digital e um número qualquer (desafio para o servidor) criptografado com a sua chave privada (E_{KRC}[n.º]), conforme observa-se na Figura 5. O campo de controle é utilizado para identificar o tipo da mensagem, neste caso uma mensagem de identificação.

Figura 5: Formato da Mensagem de Identificação

Ao receber esta mensagem, o servidor irá verificar o certificado digital do cliente, e decriptografar o número recebido com a chave pública do cliente. Em seguida, o servidor envia o seu certificado digital e o número recebido do cliente criptografado com a sua chave privada em uma mensagem de identificação. O cliente irá verificar o certificado digital do servidor e, também, se o número que recebeu criptografado com a chave privada deste é o mesmo que enviou. Se este número coincidir e o certificado digital do servidor de vídeo corresponder ao escolhido pelo cliente, este terá a garantia de estar acessando o servidor desejado.

Tendo sido os certificados verificados sem erros, o cliente irá efetuar uma requisição dos vídeos disponíveis (Requisição de Lista de Vídeos), ou então solicitar um vídeo específico (Requisição de Vídeo). Observe que no caso de um curso a distância, o próprio curso indicará o nome do vídeo a ser exibido, fazendo assim uma requisição direta do vídeo.

No caso da requisição de lista de vídeos disponíveis (Figura 6) identificada pelo campo de controle, o servidor irá verificar quais os privilégios do cliente e irá responder com a lista de vídeos. O cliente, então, irá selecionar o vídeo desejado e fazer uma Requisição de Vídeo.

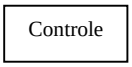

Figura 6: Formato da Mensagem de Requisição de Lista de Vídeos

Quando o servidor recebe uma requisição vídeo (Figura 7), este verifica se o usuário possui permissão de acesso ao mesmo. Em caso afirmativo, inicia o processo de transmissão do vídeo MPEG seguro. Caso contrário, envia uma mensagem de erro informando que o acesso ao vídeo não foi permitido.

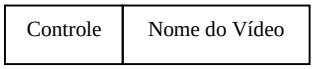

Figura 7: Formato da Mensagem de Requisição de Vídeo

É importante registrar estas transações de controle de acesso e troca de certificados em um arquivo de log do servidor de vídeo. Com estas informações é possível identificar quais são os usuários do sistema e quem acessa os vídeos disponíveis, permitindo uma posterior auditoria, cobrança, ou até mesmo uma simples verificação dos vídeos mais acessados.

5.2.2 O servidor de Vídeo: S/Server

O servidor de vídeo é responsável pela criptografia e pela transmissão do vídeo MPEG. Uma vez recebida a requisição do vídeo, o servidor deve cumprir as seguintes etapas, conforme ilustrado na Figura 8:

- criptografar o vídeo MPEG, ou localizar um já criptografado;
- montar o cabeçalho MPEG seguro;
- montar o arquivo a ser transmitido;
- transmitir o arquivo.

A criptografia do vídeo MPEG é feita utilizando um dos esquemas apresentados anteriormente, conforme o nível de segurança:

- No caso geral, utiliza-se o mais eficiente, escolha que pode ser feita tomando como base a comparação feita por [Qiao et al. (1998)], que implicaria na utilização do VEA [Qiao et al. (1998)].
- Ou então escolhe-se o algoritmo correspondente ao nível de segurança desejado, determinado quando o vídeo é cadastrado no servidor.

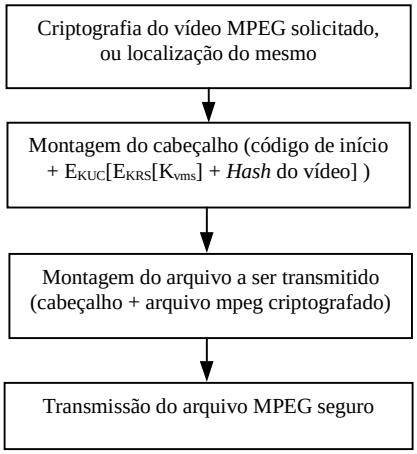

Figura 8: Fluxograma da Criptografia e Transmissão do Vídeo Seguro

A criptografia do arquivo MPEG pode ser realizada antes da transmissão do mesmo, ou somente quando o usuário solicita o vídeo (criptografia *on the fly*). Caso a criptografia seja anterior a requisição, esta será atendida

mais rapidamente, porém o nível de segurança é menor já que diversos usuários receberão vídeos MPEG seguros criptografados com a mesma chave. A criptografia no instante da requisição permite que diferentes usuários recebam arquivos MPEG seguros e com diferentes chaves de criptografia.

O cabeçalho MPEG seguro é necessário para que o visualizador (**S/Viewer**) possa reproduzir o vídeo MPEG. Este cabeçalho é composto pelo código de início de seqüência de vídeo VMS (vídeo MPEG seguro), pela chave de criptografia do vídeo VMS (K_{VMS}) e pelo *hash* do vídeo VMS.

Para garantir a confidencialidade da chave de criptografia do vídeo (K_{VMS}) e do *hash* do vídeo VMS, estas informações são criptografadas com a chave pública do usuário (K_{UC}). Além disto, para garantir a autenticidade da chave de criptografia do vídeo (K_{VMS}), esta é criptografada com a chave privada do servidor (K_{RS}). Assim o cabeçalho fica conforme observa-se na Figura 9.

Código de início	$E_{KUC}[E_{KRS}[K_{vms}]$ + *Hash* do vídeo]

Figura 9: Cabeçalho do Arquivo MPEG seguro (VMS)

Após montar o cabeçalho, este é acrescentado ao arquivo MPEG criptografado, e a transmissão do mesmo é iniciada. A transmissão do arquivo pode ocorrer de duas formas: o arquivo completo ou por *streaming*.

5.2.3 Reprodução do Vídeo MPEG: S/Viewer

O *player* seguro deve ser capaz de reproduzir vídeos MPEG padrão e vídeos MPEG criptografados. Para identificar o tipo de vídeo a ser reproduzido é necessário determinar um código de início para o vídeo MPEG criptografado, uma vez que todo vídeo MPEG padrão é identificado pelo seu código de início de seqüência.

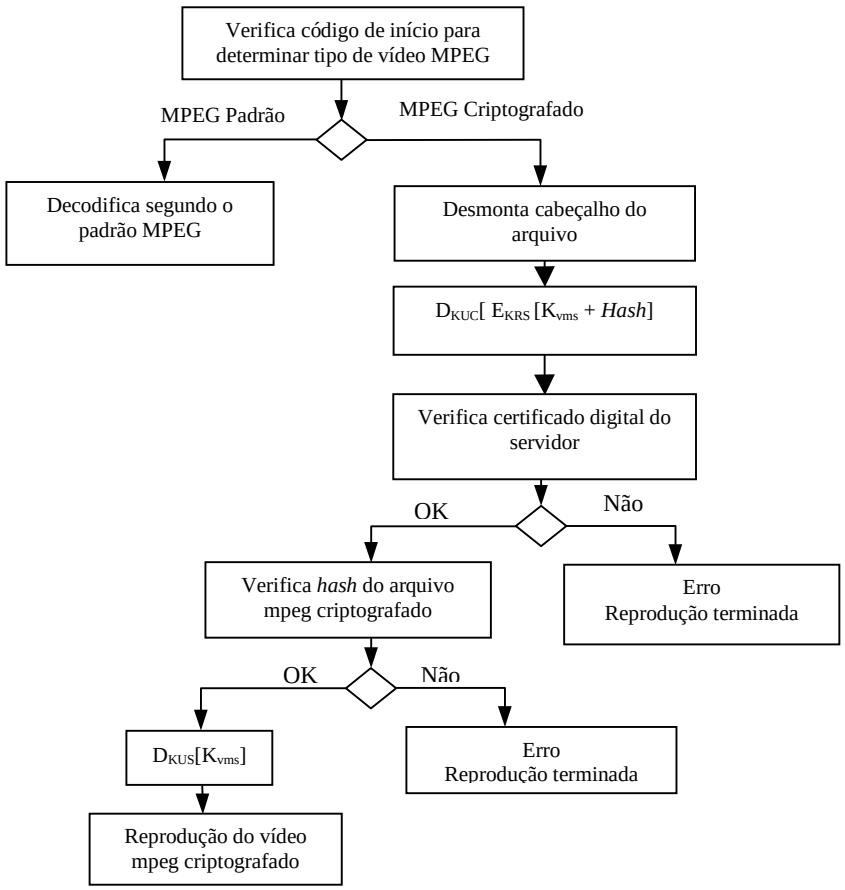

Figura 10: Fluxograma do Mecanismo de reprodução de vídeo MPEG criptografado

Assim, o primeiro passo para a reprodução de um vídeo é identificar o tipo de MPEG: padrão ou criptografado. Se for um arquivo MPEG padrão, a reprodução é executada normalmente. Caso seja um vídeo MPEG criptografado, são necessários os outros passos.

Se o vídeo a ser reproduzido for um MPEG criptografado, é necessário desmontar o cabeçalho e decriptografar as informações criptografadas utilizando para tanto a chave privada do cliente ($D_{KRC}[E_{KRS}[K_{VMS}]$ + *Hash* do vídeo], resultando em $E_{KRS}[K_{VMS}]$ e no *Hash* do vídeo).

Então, o certificado do servidor é verificado. Em seguida, calcula-se o *hash* do arquivo MPEG criptografado e compara-se com o *hash* recebido. Caso ocorra algum erro, a reprodução do arquivo é terminada.

Esta descrição considera que o *hash* do vídeo é calculado para todo o arquivo MPEG, assim é necessário ter recebido o arquivo todo para iniciar a sua reprodução. Quando considera-se distribuição de vídeo por *streaming*, torna-se necessário alterar o método de cálculo do *hash* do arquivo MPEG todo para parte do arquivo MPEG. O próximo passo é obter a chave K_{VMS} (através de $D_{US}[K_{VMS}]$) para, então, iniciar a reprodução do vídeo. A Figura 10 ilustra o processo de reprodução de um vídeo MPEG pelo *player* proposto.

Reproduzir um vídeo MPEG criptografado significa decriptografá-lo e decodificá-lo simultaneamente e em tempo real. Ou seja, o vídeo não fica disponível ao usuário decriptografado.

6 Considerações Finais

A partir da análise dos mecanismos de criptografia MPEG existentes e do levantamento dos requisitos de um MPEG *player* seguro, é possível especificá-lo e, então, implementá-lo.

O *player* MPEG seguro, S/Viewer, e o servidor , S/Server, propostos serão desenvolvidos a partir do software MPEG implementado na parte 5 do padrão ISO/IEC 13818-5 e 11172-5. Este programa foi desenvolvido em linguagem C, que será a linguagem adotada para o desenvolvimento do servidor e do *player* seguro.

Com a implementação do MPEG *player*, o próximo passo é a realização de testes para verificar o nível de segurança do mecanismo e sua velocidade de criptografia.

Um critério que deve ser utilizado para avaliar a segurança do vídeo criptografado é executá-lo em um *player* que suporte erros, e verificar se é possível identificar objetos na cena.

7 Referências Bibliográficas

Agi, I. & Gong, L. "**An Empirical Study of Secure MPEG Video Transmission**". Proceedings of the Internet Society Symposium on Network and Distributed System Security, San Diego, CA, Feb. 1996.

LeGall, D.J. **"MPEG: A Video Compression Standard for Multimedia Applications".** Communications of the ACM, Vol. 34, nº 4, April 1991.

Li, Y.; Chen, Z.; Tan, S. & Campbell, R.H. **"Security Enhanced MPEG Player"**. Proceedings of the First International Workshop on Multimedia Software Development (MMSD '96), Berlin, Germany, March 1996.

Meyer, J. & Gadegast, F. **"Sicherheitsmechanismen für Multimedia-Daten am Beispiel MPEG-I Video"**. Projektbericht, TU Berlim, 1995. http://www.mpeg1.de

Mitchel, J.L.; Pennebaker, W.B.; Fogg, C.E. & LeGall, D.J. **"MPEG Video Compression Standard"**. Chapman and Hall, 1996.

Qiao, L. and Nahrstedt, K. **"A New Algorithm for MPEG Video Encryption".** Proceedings of the First International Conference on Imaging, Science, Systems and Technology (CISST '97), Las Vegas, Nevada, July 1997.

Qiao, L. and Nahrstedt, K. "**Comparison of MPEG Encryption Algorithms**". Computer & Graphics, vol. 22, nº 4, 1998.

Spanos, G.A. & Maples, T.B. **"Performance Study of a Selective Encryption Scheme for the Security Networked, Real-Time Video"**. Proceedings of Fourth International Conference on Computer Communications and Networks, Las Vegas, Nevada, September 1995.

Spanos, G.A. & Maples, T.B. **"Security for Real-Time MPEG Compressed Video in Distributed Multimedia Applications"**. Fifteenth IEEE International Phoenix Conference on Computers and Communications, Scottsdale, AZ, March 1996.

Stallings, W. "**Criptography and Network Security, Principles and Practice**", Prentice Hall, 2nd Edition, 1998.

Tang, L. **"Methods for Encrypting and Decrypting MPEG Video Data Efficiently".** Proceedings of The Fourth ACM International Multimedia Conference (ACM Multimedia '96), Boston, MA, November 1996.