UNIP LIMEIRA – UNIVERSIDADE PAULISTA

GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

CAIO YAGO VILELA F00JED7

AS TÉCNICAS CRIPTOGRÁFICAS, CONCEITOS, USOS E APLICAÇÕES

LIMEIRA – SÃO PAULO

2019

CAIO YAGO VILELA F00JED7

AS TÉCNICAS CRIPTOGRÁFICAS, CONCEITOS, USOS E APLICAÇÕES

Relatório de Atividade Prática Supervisionada (APS) para avaliação no 2º Semestre letivo do curso de Ciência da Computação apresentado à UNIP Limeira – Universidade Paulista.

Orientador (es): Danilo Pereira e Serginho Nunes.

LIMEIRA – SÃO PAULO

2019

SUMÁRIO

1. PROPOSTA DO TRABALHO

Foi proposto que através de fontes formais de informação, aplicar à utilização do conceito de criptografia num caso específico que envolve restrição de acesso a uma área contaminada ambientalmente que contenha riscos a saúde pública: um navio foi aprendido pela guarda costeira brasileira por transportar lixo tóxico da Ásia para a região norte do Brasil. O acesso à tripulação, assim como a todo conteúdo tóxico radiativo, deverá ser controlado. Somente inspetores devidamente trajados com roupas especiais poderão adentrar no navio. Por razões legislativas o navio deve permanecer a uma distancia segura: 50 quilômetros da costa e todo e qualquer contato deverá ser realizado por meio de helicópteros, para minimizar e restringir o contato. A área do entorno num raio de 10 quilômetros está isolada.

2. OBJETIVO

O grupo deverá escolher uma técnica de criptografia e expor em sala de aula as questões relativas ao uso da mesma, tendo como cenário a rede mundial de computadores, nos seguintes aspectos: Qual a abordagem utilizada em sua concepção (estruturação, conceitos e fundamentação), Os benefícios que a mesma trouxe em relação a outras técnicas anteriores, Principais aplicações e sistemas que a utilizam ou utilizaram-na e a motivação para tal escolha da Discussão comparativa entre esta técnica e outras conhecidas utilizadas, Expor de forma analítica as especificidades de cada uma e sua utilização mais adequada, Eventuais vulnerabilidades e falhas detectadas neste tipo de técnica, Quais as melhorias futuras foram ou têm sido propostas e eventuais consequências.

3. INTRODUÇÃO

3.1 FAMÍLIA RIJNDAEL

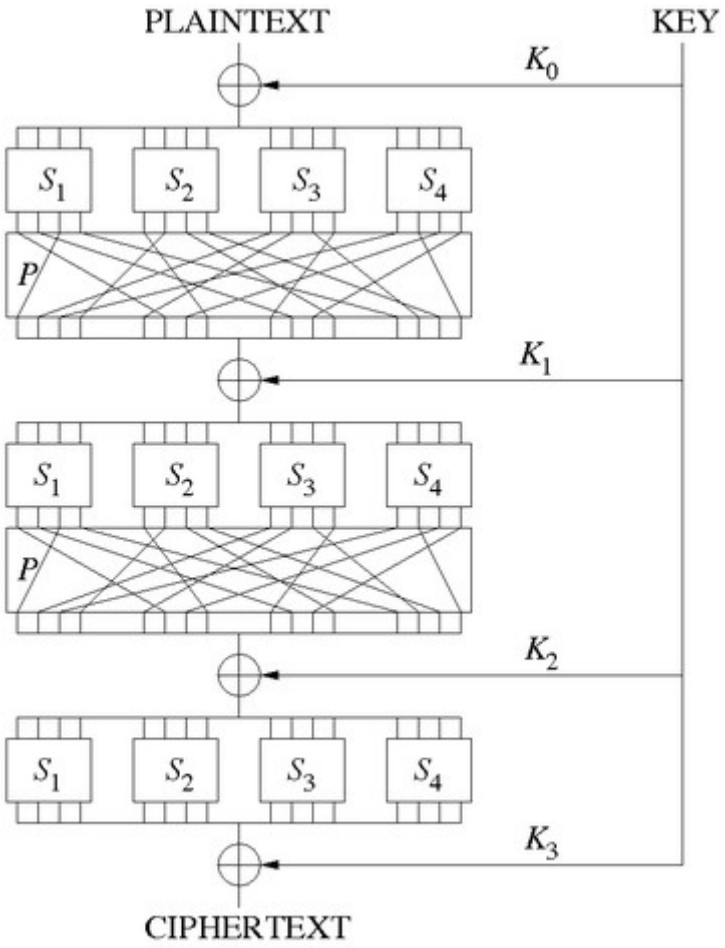
Em Criptografia, o Advanced Encryption Standard (AES, ou Padrão de Criptografia Avançada, em português), também conhecida pelo seu nome original Rijndael, é uma especificação para criptografia de dados eletrônicos estabelecida pelo Instituto Nacional de Padrões e Tecnologia (NIST) dos EUA em 2001. AES é um subconjunto de cifra de bloco da família Rijndael desenvolvida por dois criptógrafos, Vincent Rijmen e Joan Daemen, que submeteram a proposta ao NIST durante o processo de seleção AES, Rijndael é uma família de cifras com diferentes chaves e tamanhos de bloco, Para a AES, o NIST selecionou três membros da família Rijndael, cada um com um tamanho de bloco de 128 bits, mas com diferentes comprimentos de chave: 128, 192 e 256 bits. O Rijndael era um refinamento do Square, um projeto anterior de Daemen e Rijmen. O Square, por sua vez, evoluiu do Shark. Ao contrário do seu predecessor DES, o Rijndael é uma rede de permutação-substituição, não uma rede de Feistel. O AES é rápido tanto em software quanto em hardware, é relativamente fácil de executar e requer pouca memória. Apesar de ser um padrão novo de criptografia, está sendo atualmente implantado em grande escala.

3.2 AES – ESTADOS UNIDOS

O algoritmo descrito pelo AES é um algoritmo de chave simétrica, ou seja, a mesma chave é usada para criptografar e descriptografar os dados. Nos Estados Unidos, a AES foi anunciada pelo NIST como U.S. FIPS PUB 197 (FIPS 197) em 26 de novembro de 2001. Este anúncio seguiu um processo de padronização de cinco anos no qual quinze projetos concorrentes foram apresentados e avaliados, antes que a cifra Rijndael fosse selecionada como a mais adequada, AES tornou-se eficaz como um padrão do governo federal em 26 de maio de 2002, após a aprovação do Secretário de Comércio. Ela está incluída na norma ISO/IEC 18033-3. Também está disponível em muitos pacotes de criptografia diferentes e é a primeira (e única) cifra acessível publicamente aprovada pela Agência de Segurança Nacional (NSA) para informações altamente secretas quando usado em um módulo criptográfico aprovado pela NSA (veja Segurança do AES, abaixo), A AES foi adotada pelo governo dos EUA e é hoje usada no mundo todo. Ele substitui o Data Encryption Standard (DES), publicado em 1977.

4. O PROCESSO

Essa série de operações (adicionar chave, substituir, misturar/permutar) é chamada "Substitution-Permutation Network", ou SP-network. Aqui está uma representação gráfica do processo, com 3 rounds.



5. TIPOS DE CRIPTOGRAFIAS

5.1. CRIPTOGRAFIA SIMÉTRICA

Neste tipo de criptografia, uma única chave é usada para criptografia e descriptografia. É mais rápido do que o outro, mas também tem alguns inconvenientes como chave única, que é usado para criptografia e descriptografia, então quando você criptografar os dados, você tem que fornecer a mesma chave para a descriptografia e se os dados são enviados através da rede, Então no final, onde decodificação aconteceu, também precisamos saber a mesma chave. Suponha que você tem um Serviço e está fazendo a criptografia / descriptografia da mensagem com uma chave e seus muitos clientes consomem esse Serviço, então você tem que fornecer a chave para o seu cliente também. Ele precisa de confiança de nível muito alto, como você está compartilhando sua chave, o que significa seu segredo.

5.2. CRIPTOGRAFIA ASSIMÉTRICA

A criptografia assimétrica usa duas chaves para criptografia e descriptografia, onde uma chave é para criptografia e outra é para descriptografia. Criptografar mensagem por chave pública e descriptografar mensagem usando chave particular. A chave pública é usada somente para criptografia e não pode descriptografar a mensagem por chave pública, mas criptografia assimétrica é mais lenta do que outros. É muito lento, por isso não se encaixa bem para os grandes dados ainda mais do que 1kilobyte.

Principalmente dois algoritmos são usados para a criptografia assimétrica.

RSA foi descrito pela primeira vez em 1978 por Ron Rivest, Adi Shamir e Leonard Adleman e foi nomeado em seu nome RSA, que significa Ron Rivest, Adi Shamir e Leonard Adleman

DSA significa Digital Signature Algorithm.

6. DECIFRAGEM

O processo de decifrar consiste simplesmente na aplicação do inverso dessas mesmas operações, na ordem inversa naturalmente. Muitas vezes não muda nada entre cifrar e decifrar (ex.: a xor b xor b = a), em outras é necessário alguma adaptação.

Em vez de começar com estado = mensagem e terminar com cifra = estado, agora começamos com estado = cifra e terminamos com mensagem = estado.

A expansão da chave é igual; no início de cada round se adiciona a chave do round correspondente (lembrando que agora estamos fazendo os rounds de trás pra frente), e no final se adiciona a chave do primeiro round. Para se desfazer a etapa 3 (mistura de colunas) de cada round em que ela se aplica (i.e. todos exceto o primeiro e o último), usa-se a matriz inversa àquela descrita anteriormente:

$$\begin{bmatrix} a00 \\ a10 \\ a20 \\ a30 \end{bmatrix} = \begin{bmatrix} 14 & 11 & 13 & 9 \\ 9 & 14 & 11 & 13 \\ 13 & 9 & 14 & 11 \\ 11 & 13 & 9 & 14 \end{bmatrix} \begin{bmatrix} a00 \\ a10 \\ a20 \\ a30 \end{bmatrix}$$

7. O ALGORITMO AES

O AES tem um tamanho de bloco fixo em 128 bits e uma chave com tamanho de 128, 192 ou 256 bits, enquanto o Rijndael pode ser especificado com chaves e tamanhos de bloco de qualquer múltiplo de 32 bits, com um mínimo de 128 bits e um máximo de 256 bits. A chave é expandida usando-se o escalonamento de chaves do Rijndael. A maioria dos cálculos do AES é feita em um corpo finito próprio. O AES opera sobre um arranjo bidimensional de bytes com 4x4 posições, denominado de estado (as versões do Rijndael com um tamanho de bloco maior têm colunas adicionais no estado). Para criptografar, cada turno do AES (exceto o último) consiste em quatro estágios.

AddRoundKey - cada byte do estado é combinado com a subchave própria do turno (RoundKey); cada subchave é derivada da chave principal usando o algoritmo de escalonamento de chaves.

SubBytes - é uma etapa de substituição não linear onde cada byte é substituído por outro, de acordo com uma tabela de referência.

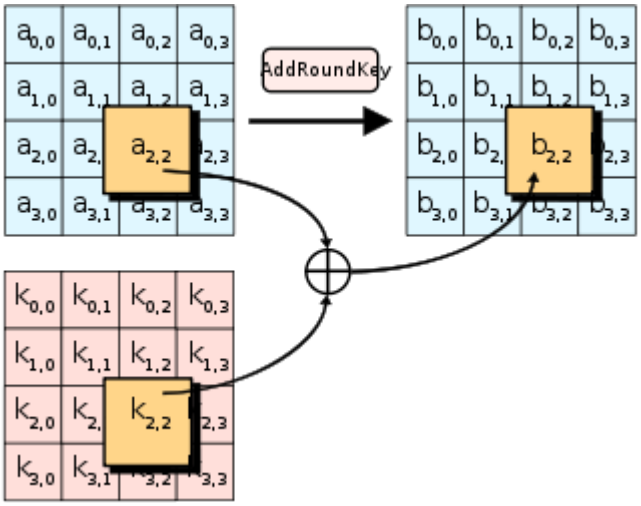
ShiftRows - é uma etapa de transposição onde cada fileira do estado é deslocada de um determinado número de posições.

MixColumns - é uma operação de mescla que opera nas colunas do estado e combina os quatro bytes de cada coluna usando uma transformação linear.

O turno final substitui o estágio de MixColumns por um novo estágio de AddRoundKey.

7.1. ETAPA ADDROUNDKEY

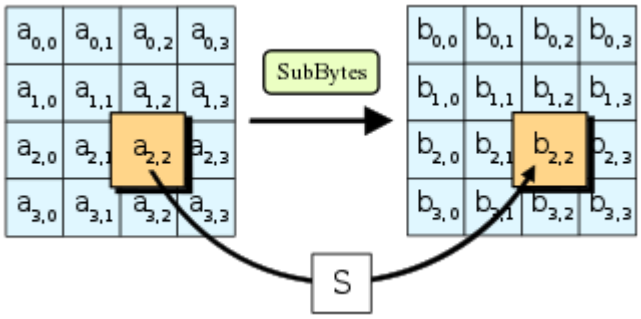
Na etapa de AddRoundKey, a sub-chave é combinada com o estado. Para cada turno, uma sub-chave é derivada da chave principal usando o escalonamento de chaves do Rijndael; cada sub-chave é do mesmo tamanho que o estado. A sub-chave é somada combinando cada byte do estado com o byte correspondente do sub-chave, utilizando a operação XOR bit a bit.



(Na etapa de AddRoundKey, a sub-chave é combinada com o estado)

7.2. ETAPA SUBBYTES

Na etapa de SubBytes, cada byte no arranjo é atualizado usando uma S-box de 8 bits. Para evitar os ataques baseados em propriedades algébricas simples, a S-box é construída combinando-se uma função inversora com uma transformação afim invertível. A S-box é escolhida também de forma a evitar qualquer ponto fixo.

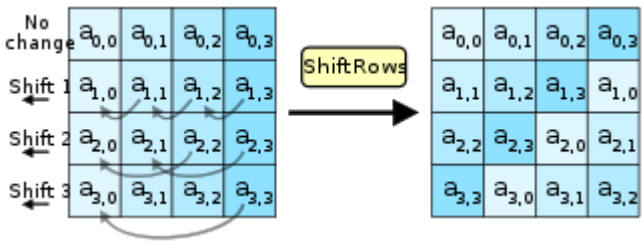


(Na etapa de SubBytes, cada byte no arranjo é atualizado usando uma S-box de 8 bits.)

7.3. ETAPA SHIFTROWS

A etapa de ShiftRows opera sobre as linhas do estado, deslocando os bytes em cada linha de um determinado número de posições. No AES, a primeira linha fica inalterada. Cada byte da segunda linha é deslocado à esquerda de uma posição. Similarmente, a terceira e quarta fileiras são deslocadas de duas e de três posições respectivamente. Para o bloco de bits de tamanho 128 e 192 bits, o padrão de deslocamento é mesmo. Desta forma, cada coluna do estado ao fim da etapa de ShiftRows fica composta por bytes de todas as coluna do estado da entrada.

No caso de blocos de 256 bits, a primeira fileira fica inalterada, deslocando-se a segunda, terceira e quarta fileiras. O deslocamento é de 1 , 2 e 4 bytes respectivamente - embora esta mudança se aplique somente ao Rijndael quando usado com um bloco de 256 bits, o que não ocorre no AES.

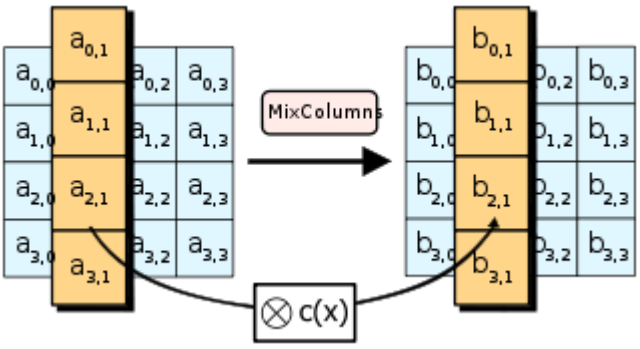


(A etapa de ShiftRows opera sobre as linhas do estado; deslocando ciclicamente os bytes em cada linha de um determinado número de posições.)

7.4. ETAPA MIXCOLUMNS

Na etapa de MixColumns, os quatro bytes de cada coluna do estado são combinados usando uma transformação linear invertível. Junto com o ShiftRows, o MixColumns fornece difusão à cifra. Cada coluna é tratada como um polinômio com coeficientes em {\displaystyle GF(2^{8})}GF(2^8) e é então multiplicado em módulo {\displaystyle x^{4}+1}x^4 + 1 pelo polinômio fixo {\displaystyle c(x)=3x^{3}+x^{2}+x+2}c(x) = 3x^3 + x^2 + x + 2.

A etapa de MixColumns pode também ser vista como uma multiplicação matricial no corpo finito de Rijndael.



(Na etapa de MixColumns, os quatro bytes de cada coluna do estado são combinados usando uma transformação linear invertível.)

8. CÓDIGO DO PROJETO

Para o Código na prática utilizamos uma base de *JOSÉ MORAIS do* Site *vidadesilicio.com* criado em 14 de julho de 2017.

```
#include <AES.h>//Biblioteca do AES.

AES aes;//Cria a classe aes.

byte key[16], out[16], inp[32];//Cria arrays (vetores) para a chave, input e output de dados.

const char pass[] = "abc";//Define a chave usada, neste exemplo usamos AES128, então precisa ser <= 16 Bytes.

void setup()

{

  Serial.begin(115200);//Habilita a serial.

  Serial.println();//Limpa o monitor.

    enc128("vida de silicio", 1);//Faz a função de encriptação e retorna o HEX encriptado.

}

void loop()

{

}

void enc128(const char txt[], bool db)//Argumentos: (texto e debug)

{
```

```
if (strlen(pass) > 16)//Verifica se a chave tem o tamanho limite de 16 caracteres.
{
  if (db == true)
  {
    Serial.println("Chave para AES128 <= 16 Bytes");
  }
  return;//Se a chave for maior, irá sair da função.
}
if (strlen(txt) > 16)//Verifica se o texto tem o tamanho limite de 16 caracteres.
{
  if (db == true)
  {
    Serial.println("Frase/numero para AES <= 16 Bytes / bloco");
  }
  return;//Se o texto for maior, irá sair da função.
}
for (byte i = 0; i < strlen(pass); i++)//Adiciona a chave(pass) na array key.
{
  key[i] = pass[i];
}
for (byte i = 0; i < strlen(txt); i++)//Adiciona o texto na array input.
{
  inp[i] = txt[i];
```

```
  }
  //Adiciona a chave ao algoritimo.
   if (aes.set_key(key, 16) != 0)//Verifica se a chave esta correta, caso nao, sairá da função.
  {
    if (db == true)
    {
      Serial.println("Erro ao configurar chave");
    }
    return;//Sai da função
  }
  //Faz a encriptação da array INPUT e retorna o HEXA na array OUTPUT.
   if (aes.encrypt(inp, out) != 0)//Verifica se a encriptação esta correta, se não, sairá da função.
  {
    if (db == true)
    {
      Serial.println("Erro ao encriptar");
    }
    return;//Sai da função
  }
  if (db == true)//Se o debug estiver on (1), irá mostrar o HEXA no serial monitor.
  {
```

```
    for (byte i = 0; i < 16; i++)
    {
      Serial.print(out[i], HEX);
      Serial.print(" ");
    }
    Serial.println();
  }
  aes.clean();//Limpa a chave e residuos sensiveis da encriptação.
}
```

9. COMPARAÇÃO AES COM OUTROS ALGORITMOS

Algoritmo	*Tamanho da Chave*	*Nº de Rodadas*	*Operações Matemáticas*
AES	128, 192 ou 256 bits	10, 12, 14	XOR, S-Boxes fixas
DES	56 bits	16	XOR, S-Boxes fixas
3DES	112 ou 168 bits	48	XOR, S-Boxes fixas
IDEA	128 bits	8	XOR, adição, multiplicação
Blowfish	Variável ate 448 bits	16	XOR, S-Boxes variáveis, adição
RC5	Variável ate 2048 bits	Variável ate 255	Adição, Subtração, XOR, rotação
CAST-128	40 ate 128 bits	16	Adição, subtração, XOR, rotação, S-Boxes fixas

10. RESOLUÇÃO

10.1. VANTAGEM

Pode rodar bem rapidamente em relação a outros algoritmos, Pode ser implementado em um SmartCard usando pouco código e memória, Algumas funções podem ser feitas em paralelo assim tornando o processo mais rápido, pode se fazer a substituição dos bytes usando a S-box, vários de uma vez e não sequencialmente e ainda, a expansão da chave pode ser feita enquanto se executa as funções que não dependem dela como a ByteSub() ou ShiftRow(). Como a encriptação não emprega operações aritméticas, não exige muito poder de processamento. Simplicidade do projeto, Não usa elementos já previamente processados, por ex. a 9a rodada não necessita de nenhum elemento das rodadas anteriores a não ser única e exclusivamente do State da 8a . O algoritmo não baseia sua segurança ou parte dela em interações obscuras e não bem

compreendidas entre operações aritméticas, não permitindo assim, espaço para esconder um trap-door.

9.1.1. POSSÍVEIS MODIFICAÇÕES

O projeto permite a especificação de variantes com o comprimento do bloco e da chave, ambos variando de 32 em 32 bits no intervalo de 128 até 256 bits, Embora o número das rodadas de seja fixo na especificação, pode ser modificado como um parâmetro caso haja problemas de segurança.

9.2 DESVANTAGEM

As limitações ficam por conta da inversa, A inversa é menos recomendável de ser implementada num SmartCard pois precisa de mais código e mais processamento, Mesmo assim se comparado a outros algoritmos ela bem rápida. Em software a encriptação e sua inversa empregam códigos diferentes e/ou tabelas. Em hardware a inversa pode usar apenas uma parte do circuito usado no processo de encriptação.

11. CONCLUSÃO

Esse trabalho apresentou o algoritmo que e o atual padrão de criptografia dos EUA, destinado a ser lançado sob uma base global, não exclusiva e livre de royalties, o algoritmo do AES é eficiente em termos computacionais e de memória.

Com a evolução contínua da tecnologia, a prevalência de ciberataques permanece crescendo. Atualmente, não existe nenhum método conhecido de violação da criptografia AES, o que a torna uma grande força propulsora de segurança, essencial para proteger suas informações e reduzir o risco de ataques. A criptografia AES já está integrada a diversos sistemas de software e hardware. E, caso venha a ser adotada plenamente, seu potencial parece quase ilimitado.

12. BIBLIOTECA

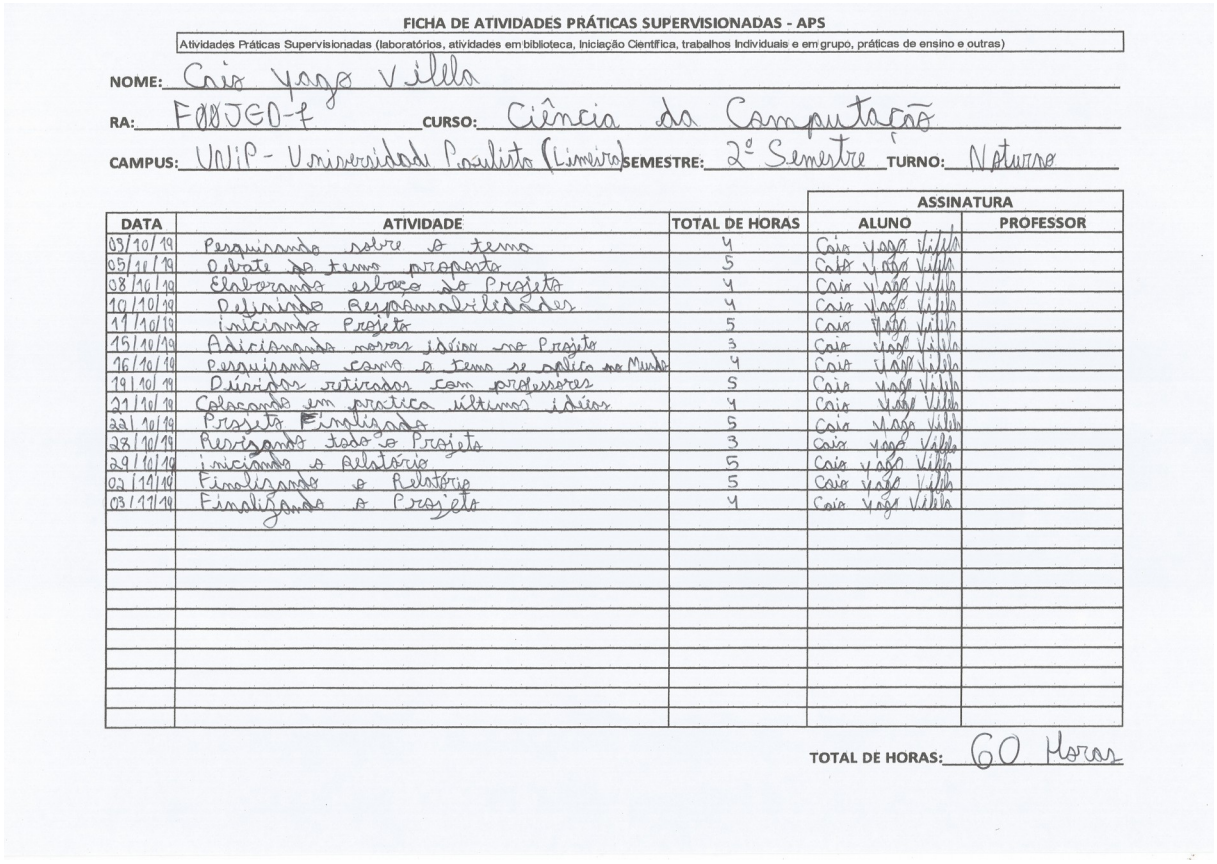
WIKIPÉDIA, A ENCICLOPÉDIA LIVRE. Wikipédia, a enciclopédia livre. Wikipédia, a enciclopédia livre. Disponível em: <https://pt.wikipedia.org/wiki/Advanced_Encryption_Standard#:~:text=Em%20Criptografia%2C%20o%20Advanced%20Encryption,NIST)%20dos%20EUA%20em%202001.>. Acesso em: 18 Outubro 2019.

EMBARCADOS. Segurança da Informação - Criptografia AES. Disponível em: <https://www.embarcados.com.br/criptografia-aes/>. Acesso em: 20 Outubro 2020.

GTA. Algoritmo de criptografia AES. feito por: Leopoldo A. P. Mathias. Disponível em: <https://www.gta.ufrj.br/grad/05_2/aes/>. Acesso em: 05 Outubro 2019.

STACKOVERFLOW. Como funciona o algoritmo de criptografia AES. Disponível em: <https://pt.stackoverflow.com/questions/43492/como-funciona-o-algoritmo-de-criptografia-aes>. Acesso em: 27 Outubro 2020.

13. FICHA APS



FICHA DE ATIVIDADES PRÁTICAS SUPERVISIONADAS - APS

Atividades Práticas Supervisionadas (laboratórios, atividades em biblioteca, Iniciação Científica, trabalhos individuais e em grupo, práticas de ensino e outras)

NOME: Caio Yago Villa

RA: F88JGD-7 CURSO: Ciência da Computação

CAMPUS: UNIP - Universidade Paulista (Limeira) SEMESTRE: 2º Semestre TURNO: Noturno

DATA	ATIVIDADE	TOTAL DE HORAS	ASSINATURA	
			ALUNO	PROFESSOR
03/10/19	Pesquisando sobre o tema	4	Caio Yago Villa	
05/10/19	Debate do tema proposto	5	Caio Yago Villa	
08/10/19	Elaborando esboço do Projeto	4	Caio Yago Villa	
10/10/19	Definindo Responsabilidades	4	Caio Yago Villa	
11/10/19	Iniciando Projeto	5	Caio Yago Villa	
15/10/19	Adicionando novas ideias no Projeto	3	Caio Yago Villa	
16/10/19	Pesquisando como o tema se aplica no Mundo	4	Caio Yago Villa	
19/10/19	Duvidas retiradas com professores	5	Caio Yago Villa	
21/10/19	Colocando em pratica ultimas ideias	4	Caio Yago Villa	
22/10/19	Projeto Finalizado	5	Caio Yago Villa	
28/10/19	Revisando todo o Projeto	3	Caio Yago Villa	
29/10/19	Iniciando o Relatorio	5	Caio Yago Villa	
02/11/19	Finalizando o Relatorio	5	Caio Yago Villa	
03/11/19	Finalizando o Projeto	4	Caio Yago Villa	

TOTAL DE HORAS: 60 Horas