Aula 2 TP – 11/Fev/2019

Grupo 3

1. *Números aleatórios/pseudoaleatórios*

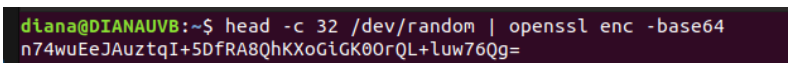
- ***Experiência 1.1***

Executando o seguinte comando, que gera 1024 bytes pseudoaleatórios: *openssl rand -base64 1024*, obtemos o seguinte resultado:

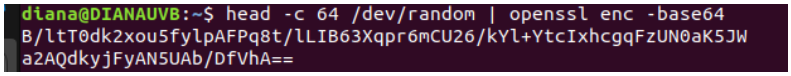
- ***Pergunta 1.1***

Testando os seguintes comandos, vamos obter os seguintes resultados:

a) *head -c 32 /dev/random | openssl enc -base64*



b) *head -c 64 /dev/random | openssl enc -base64*



c) *head -c 1024 /dev/random | openssl enc -base64*

d) *head -c 1024 /dev/urandom | openssl enc -base64*

Estes testes permitem-nos concluir que é possível gerar números pseudoaleatórios através do */dev/random* e */dev/urandom*. No caso do */dev/random*, este faz o *polling* do sistema em execução para ter entropia suficiente. Se não tiver entropia suficiente este ficará bloqueado, ou seja, quando o conjunto de entropia disponível está esgotado, será necessário aguardar até que a entropia adicional seja reunida para obter dados aleatórios. No nosso caso, como o fluxo de entropia do sistema se encontrava alto, devido à quantidade de aplicações a correr, os resultados foram obtidos rapidamente. No caso do /dev/urandom, este não ficará bloqueado, uma vez que ele reutilizará a entropia que existe no sistema, permitindo que haja uma resposta rápida. Contudo, será diminuída a qualidade da aleatoriedade ao longo do tempo, mesmo assim ainda considerado um PRNG criptograficamente seguro.

- ***Pergunta 1.2***

Para instalar a package *haveged* na máquina virtual executamos o seguinte comando: sudo apt-get install haveged.

Ao testar novamente os seguintes comandos, onde obtemos 1024 bytes pseudoaleatórios do sistema e os apresentaram em base64:

a) *head -c 1024 /dev/random | openssl enc -base64*

b) *head -c 1024 /dev/urandom | openssl enc -base64*

Analisando os resultados obtidos nas alíneas a) e b), vemos que o tempo de execução dos resultados é idêntico. Contudo, isto é conseguido devido à instalação do daemon de entropia adaptado do algoritmo HAVEGE. Esta package serviu para ultrapassar as condições de baixa entropia no dispositivo aleatório do Linux.

2. Partilha/Divisão de segredo (Secret Sharing/Splitting)

- ***Experiência 2.1***

Através do ***genSharedSecret.php*** codificamos o segredo “EngenhariaSegurança” em 5 partes.

```
root@CSI:~/Desktop# php genSharedSecret.php "EngenhariaSegurança" 5
Codigo 0: 1011000011011011111111011100001001110101001010100101100011010100110100
1001001000010100000000001001000010110001111100010000110111000001101010100100100
1001100101
Codigo 1: 1000010011010001110110011001100110110011100010100101111111010111000010
1000111001011000101000011000010001111001001110110111110101100111100010011110100
0110101110
Codigo 2: 0000111001100000101100110101010111011111000100100100110100110011110000
1100110011010110101111001110000011000010100111011011001011111010000110111101 1
0001001111
Codigo 3: 0111110100110110100111111000110111110000001011100111111001001000010100 1
1111001110011110011000110010111001011101011111010001011101011101001010000011101 0
0101011001
Codigo 5: 0000001000110010011010011010000000001000111110000011111010101111101001
0111101101010000101001111000001110001010011101100100100111111111111111000000011010
0110111100
```

Na imagem seguinte, primeiro usamos o ***reconstroiSecret.php*** para reconstruir o segredo juntando as 5 partes obtidas anteriormente, e obtivemos sucesso.

De seguida, tentamos reconstruir apenas com 4 partes e posteriormente com 6 (sendo que repetimos uma das partes). Em ambos os casos não obtivemos sucesso.

```
root@CSI:~/Desktop# php reconstroiSecret.php 1011000011011011111111011100001001110101001(
1101110000011010101001001001100110010 1000010011010001110110011001100110110011100010100
1001111000100111101100011010111 0000111001100000101100110101010111011111100010010010011(
0100001101111011000100111 0111110100110110100111111000110111110000001011100111111001001(
0000111010010101100 0000001000110010011010011010000000001000111110000011111010101111101
11010110111100
Segredo: EngenhariaSegurança
root@CSI:~/Desktop#
root@CSI:~/Desktop#
root@CSI:~/Desktop# php reconstroiSecret.php 1011000011011011111111011100001001110101001(
1101110000011010101001001001100110010 1000010011010001110110011001100110110011100010100
1001111000100111101100011010111 0000111001100000101100110101010111011111100010010010011(
0100001101111011000100111 0111110100110110100111111000110111110000001011100111111001001(
0000111010010101100 1
Segredo: G\f_ i\F##
root@CSI:~/Desktop# php reconstroiSecret.php 1011000011011011111111011100001001110101001(
1101110000011010101001001001100110010 1000010011010001110110011001100110110011100010100
1001111000100111101100011010111 0000111001100000101100110101010111011111100010010010011(
0100001101111011000100111 0111110100110110100111111000110111110000001011100111111001001(
0000111010010101100 0000001000110010011010011010000000001000111110000011111010101111101
11010110111100 0000001000110010011010011010000000001000111110000011111010101111101001
10111100
Segredo: G\f_ i\F##
```

Podemos concluir assim, que para reconstruir o segredo precisamos das N partes em que foi dividida a mensagem quando a codificamos. Usando menos ou mais partes não iremos obter o segredo.

- ***Experiência 2.2***

Como se vê na imagem seguinte, usamos o algoritmo ***shares.pl*** para codificar o segredo "EngenhariaSegurança" em 7 partes, sendo que para depois reconstruir o segredo através do ***reconstruct.pl*** apenas precisamos de 3 dessas 7 partes.

```
root@CSI:~/Desktop# echo "EngenhariaSegurança" | perl shares.pl 3 7
3:1:10dd0553107a6a93b6babe942dfbf547e8da0b91:
3:2:1935b54d78df4452f7c4c9ef37a6dacd5ba747cf:
3:3:60787553a596f0b02b7f7475857721f2c92a5a1a:
3:4:e5a5466597a06cac54ecc027166eccb630644474:
3:5:a7bc28834efdba467109ac05ec8bd919925405dc:
3:6:a7bd1badcbacd97f82d9380f04ce481ceefb9e51:
3:7:e5a81fe30caec9568759654561361abf43570dd5:
root@CSI:~/Desktop# perl reconstruct.pl <<EOF
> 3:1:10dd0553107a6a93b6babe942dfbf547e8da0b91:
> 3:3:60787553a596f0b02b7f7475857721f2c92a5a1a:
> 3:5:a7bc28834efdba467109ac05ec8bd919925405dc:
> EOF
EngenhariaSegurança _
```

De seguida, tentámos reconstruir o segredo apenas com 2 partes, o que não foi possível, isto porque 2 partes não são suficientes, porque anteriormente dissemos que seriam necessárias 3 partes.

Tentámos também reconstruir com 4 partes e neste caso já obtivemos sucesso, pois como eram necessárias apenas 3, o algoritmo ignorou a última parte dada.

```
root@CSI:~/Desktop# perl reconstruct.pl <<EOF
> 3:1:10dd0553107a6a93b6babe942dfbf547e8da0b91:
> 3:2:1935b54d78df4452f7c4c9ef37a6dacd5ba747cf:
> EOF
too few shares at reconstruct.pl line 77, <STDIN> line 2.
root@CSI:~/Desktop# perl reconstruct.pl <<EOF
> 3:5:a7bc28834efdba467109ac05ec8bd919925405dc:
> 3:6:a7bd1badcbacd97f82d9380f04ce481ceefb9e51:
> 3:7:e5a81fe30caec9568759654561361abf43570dd5:
> 3:2:1935b54d78df4452f7c4c9ef37a6dacd5ba747cf:
> EOF
Ignoring share 2...
EngenhariaSegurança
```

- ***Pergunta 2.1***

Primeiramente geramos uma private-key.pem e um certificado.

```
root@CSI:~/Desktop# openssl genrsa -aes128 -out private-key.pem 1024
Generating RSA private key, 1024 bit long modulus
.........................+++++
......+++++
e is 65537 (0x010001)
Enter pass phrase for private-key.pem:
Verifying - Enter pass phrase for private-key.pem:
```

```
root@CSI:~/Desktop# openssl req -key private-key.pem -new -x509 -days 365 -out certificado.crt
Enter pass phrase for private-key.pem:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:PT
State or Province Name (full name) [Some-State]:MINHO
Locality Name (eg, city) []:BRAGA
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Engenharia de Seguranca
Organizational Unit Name (eg, section) []:Grupo3
Common Name (e.g. server FQDN or YOUR name) []:grupo3
Email Address []:
```

A:

Para dividir o segredo "Agora temos um segredo extremamente confidencial" em 8 partes, com quorom de 5, foi usado o comando dado:

- *python createSharedSecret-app.py number_of_shares quorum uid private-key.pem*

onde number_of_shares = 8, quorum = 5, uid = 1 e a private-key.pem gerada anteriormente.

```
root@CSI:~/Desktop# python createSharedSecret-app.py 8 5 1 private-key.pemPrivate key passphrase: segredo
Secret: Agora temos um segredo extremamente confidencial
Component: 1
eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjEtYzJiOTZmMzBlODg1ZmFkNDViZTA3ZWU5YjNkYTAwY2MwOTBkNDdhNzRiY2ZmN2YxOT
JkZThiYjk0MjcxODk0MDU0ZjYwMDM3NDkwM2IwYzcyNjQzY2Y5NTRlYzc3YThjIiwgIjEiLCA1LCA4LCAiMWU5YmIwYTUwYjNkM2EyYmRkMWZ
iNWNhOTljZGVmNjAzYzU2Y2I2NDdjMzI0YzdkNDFlNzIwMmQ1NGI5NjNkNiJdfQ.R1U4w-wHlU86rfU9MlPfPBhBZG1i-PROf6WIsdEsqHzZ1
XmUg08odzgfXRwnYAW_yc9-cYOXPSara05VU3O6FOAritJnQYKOssfWZIlC_o5doWt3cBl0h8HkNKhTRwhLD-8kcXJEHKbW9LmEBoSomC-YdL
18dNLSREmemsQBy1U
Component: 2
eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjItMWYxOTM3OTRkMTQ3ZjBjZGMyMTNhZWZkYjZiZmExYWVjYzUzYWY2OGFlMDA3YmEwZG
U3MTkzMDMwODlmNzVjOWEwNzU4ZWFiZTJjMTExMjljNzFiNDkxNTg0OGE2ZjQ0IiwgIjEiLCA1LCA4LCAiYTlhZGY4NDQyMDQ2MjBkZGY3ODA
yMTA1NGMyYjAzYmRkMDAyNTAzYzdmYzg0ZWI1NjZhMTllNjAwNGZkY2E0OCJdfQ.LDCZ_XhjZOS7riS66Iygu3EYeAdxR1zj9Ib57p5ESDr93
5ITivf8xOTpRRq5VLKUGrWnQ4HiRySpzK0xYDgB9raTvw1ENt8WtulmcF3k-0NzUdUhjVeaYch-Swpbs1Ac8CnOWa7rM6JY0457B0KoJ_mBdi
e9Nydyn1SmKoPdZmY
Component: 3
eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjMtYjFlY2I3NDNkMjVhNzA3OTg4YTE5YmZhYzEyYjI3OGIxMGI1MjcxNDIwYjllNTI1Mj
JhZTg5MzNhOWZhYmFkYjI0NWM0ZDQyY2UyMzAyODlmYzljNWVkY2YzMWNlNGI4IiwgIjEiLCA1LCA4LCAiMzlhNGE0ZDIzYWUwNWE5MzNiYTg
2YjkyM2NkY2M3OTI0NDlkYTg5OWFjZDhlZjJiN2RhM2JlZTdhZDlmNzExMSJdfQ.d2dgbPqdqGdbyqfavwL8QIRVdui9Q8mnt-mn02sAQD799
RqzWL_4pKSQH9bcDVYlWo3uZZ0cWpkqIH9Y43tOH--vFKY53iHIEHZ0fE3myLnbMYy1SW7PcySEHFc87UOghR20zhuMkQ8QVujOO4tmTyCKE8
MkXn9rJ_wjTbSn7bk
Component: 4
eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjQtMTVjNjgwNWVhNjIxODAwMWM1OTQ2NTU2MDdmZTIzNmI5ZjIwY2Y3OWQ2MTlhMzgyZj
dkNzg5MzAzZGJjN2RhZDEzNWZjYzA4ZGQ3NTI4ZGJkYmY5MTI0ODRkZThiMTBlIiwgIjEiLCA1LCA4LCAiMWY1MGI4MjgyYWEyMDgwYWE1ZDI
1YTViZDFjYTdlNTQ1YjhkZTRmZmJlZTMzMzg4YTJhYTQwY2I3YjkyZDFkNSJdfQ.fNdfXVcuhsOt24tH9se0OR35gkxFT7tfezwA8gwSmni86
aI3S6dRK9LnllLSZHp7OfWhHgyzkj_gUWNJ4mq12w2wlwlwBQUXs43UligiBp8Ia5bam82lk2cJw-UzXbmJPuya2FLt8QmIrqvdWXrJ3_olBI
DIwuZJQ-qgBbBWktk
Component: 5
eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjUtMjQ2NWM4ODEwYTcxMTE5MmJjNjdiMTViYjFlZTkyZjZhN2ZhNmNjZjljYTE1YjExYz
ZlZmM3ZDIwODQ3ODBkNWZmODFlZTUzMTdkMzU5MGJkZWUzZDkyMzBkNTZlMzhkIiwgIjEiLCA1LCA4LCAiZDYyYjI1MDc4OTg1ODU0YjFjZmV
hMGE3MGI2OTY0ZjM3Yjk1NWJjYTM5MmQwMzcwZTQ2MTE5NWNjZjYwMzgxOCJdfQ.K0zBft1Gtmbdls-Wd3ggRPW4UHueDWlT20_JPgfCGKQjJ
qTsRJLNYHP-aLPiz_dGIIcF6naKfddj43ZiBdqqY19r_F9qDUNF3g7TF6p-6HjmCrF7jiu9j5v-6vS0kYtBr3pgJ6f7siOuH_Zhf8Y1Byc4lG
ECOyBPhWQ9IuNDHLw
Component: 6
eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjYtZjZiNjY4YzFjMDhkMDM1YWRlMDhhZDJlZDc4NmUwNmRjMzFhNjZiMGRmMzZlNTdiYz
liYjkzZTY3YjI4MjQ0YWRhMTBlYWJlYjkyOTk4ZmVlMzhmOWRhNzZlY2ZiZjMzIiwgIjEiLCA1LCA4LCAiM2ZjOTgxOWNmYzBkOTI0ZTkzYmV
mZmI4Y2M4MjlkNmRjMjliNDZmZTc5MjI3ZWM4M2I2NGQ1YThmYzI2ZWYxOCJdfQ.A_PcHjN-wPGUCVQ_KpuUfW_pzCSq9PchOlSMyxkR5QJtD
9b20bbVCt_bOHJ5seX-J-XOVwDHedW6padbYcE6VLa4QJmXix55VsmSmqhrbr8va5EKmvWZk0fFLBYVx1igWcTfVeF1iJ0csegxA4KcIi7K9E
JJof6-KB6gybENwto
Component: 7
eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjctZTRkMGRkYjI4ZDI5MWY4YWM4ZDYwY2M5ODMyNWUyYWRlZmNkYzgxZWE1MjQ1MWJlMG
FjMDU2MmYzNTE0NjU1ZWYzYTc0YjMxMzIzNDFmOTIyY2FmYmU3ZTc0YmFiYWI1IiwgIjEiLCA1LCA4LCAiNTk2ZmZjMWJjMmZkZDFlNDM1NTY
wMmYxZTRkN2JiMWI0YzRiMWJkMDk3ZWRjMjkxNGQ2NDIzM2Y1ZTQwNTc2YiJdfQ.ImU6Up_P6Oc6dRLrxUKc2YTmTRW_bWMOJ2NsqYbUvQ7kH
Av30yQoOTjZlqhhk7W01cBZKvlLEEzlG0mwb9Tf4ORpH3PVoH8LRVz_b_TekkvZbq_rcyYNsNIgX7opBTN8KOOuhoan8oi9mJ09Tri3cq89fN
4u1STKGSHw7jrr3rA
Component: 8
eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjgtODVmYTQ3NjAzODY5MWM1NTQ4YTAwYWZjYjBmZWRkMmY5NGQ2M2Y3ZjkyMTdlNDI4Nj
U0NDkxNjIwMWViZGVjM2ZjMmJlY2Q4Mjg3ZjU0OWI2MTc4MGViZmU2N2U4NDFiIiwgIjEiLCA1LCA4LCAiMmNkNDRjOThkMDc3MGIzMWViNmM
5YmZmOGEwMGZiYmViZTBkOTA0NmQzYWQxNjJhNzAwMzEyYzJmMmU4YWEwNiJdfQ.X2R9Qy4dvTDlE8heVlWEYuiJxYdVDoJ1tCGt2F_9074cb
oBRQoeRirx46Ox5jdV-L95aD97Z2HQz3fz2GXymIbc10tkVa880pQHkGuu1ed7YBqA7cDw1WrBIqBfbWrqna-j4A6fMYeHG_37QX3Q6bq3f7z
VsTQt3t5LEThSuU0I
```

B:

Ao usarmos o ***recoverSecretFromComponents-app.py,*** como mostra na figura em baixo, apenas precisamos de 5(quorum) partes para reconstruir o segredo.

```
root@CSI:~/Desktop# python recoverSecretFromComponents-app.py 5 1 certificado.crt
Component 1: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjEtYzJiOTZmMzBlODg1ZmFkNDViZTA3ZWU5YjNkYTAwY2MwOTBkNDdhN
zRiY2ZmN2YxOTJkZThiYjk0MjcxODk0MDU0ZjYwMDM3NDkwM2IwYzcyNjQzY2Y5NTRlYzc3YThjIiwgIjEiLCA1LCA4LCAiMWU5YmIwYTUwYj
NkM2EyYmRkMWZiNWNhOTljZGVmNjAzYzU2Y2I2NDdjMzI0YzdkNDFlNzIwMmQ1NGI5NjNkNiJdfQ.R1U4w-wHlU86rfU9MlPfPBhBZG1i-PRO
f6WIsdEsqHzZ1XmUg08odzgfXRwnYAW_yc9-cYOXPSara05VU3O6FOAritJnQYKOssfWZIlC_o5doWt3cBl0h8HkNKhTRwhLD-8kcXJEHKbW9
LmEBoSomC-YdL18dNLSREmemsQBy1U
Component 2: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjUtMjQ2NWM4ODEwYTcxMTE5MmJjNjdiMTViYjFlZTkyZjZhN2ZhNmNjZ
jljYTE1YjExYzZlZmM3ZDIwODQ3ODBkNWZmODFlZTUzMTdkMzU5MGJkZWUzZDkyMzBkNTZlMzhkIiwgIjEiLCA1LCA4LCAiZDYyYjI1MDc4OT
g1ODU0YjFjZmVhMGE3MGI2OTY0ZjM3Yjk1NWJjYTM5MmQwMzcwZTQ2MTE5NWNjZjYwMzgxOCJdfQ.K0zBft1Gtmbdls-Wd3ggRPW4UHueDWlT
20_JPgfCGKQjJqTsRJLNYHP-aLPiz_dGIIcF6naKfddj43ZiBdqqY19r_F9qDUNF3g7TF6p-6HjmCrF7jiu9j5v-6vS0kYtBr3pgJ6f7siOuH
_Zhf8Y1Byc4lGECOyBPhWQ9IuNDHLw
Component 3: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjYtZjZiNjY4YzFjMDhkMDM1YWRlMDhhZDJlZDc4NmUwNmRjMzFhNjZiM
GRmMzZlNTdiYzliYjkzZTY3YjI4MjQ0YWRhMTBlYWJlYjkyOTk4ZmVlMzhmOWRhNzZlY2ZiZjMzIiwgIjEiLCA1LCA4LCAiM2ZjOTgxOWNmYz
BkOTI0ZTkzYmVmZmI4Y2M4MjlkNmRjMjliNDZmZTc5MjI3ZWM4M2I2NGQ1YThmYzI2ZWYxOCJdfQ.A_PcHjN-wPGUCVQ_KpuUfW_pzCSq9Pch
OlSMyxkR5QJtD9b20bbVCt_bOHJ5seX-J-XOVwDHedW6padbYcE6VLa4QJmXix55VsmSmqhrbr8va5EKmvWZk0fFLBYVx1igWcTfVeF1iJ0cs
egxA4KcIi7K9EJJof6-KB6gybENwto
Component 4: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjctZTRkMGRkYjI4ZDI5MWY4YWM4ZDYwY2M5ODMyNWUyYWRlZmNkYzgxZ
WE1MjQ1MWJlMGFjMDU2MmYzNTE0NjU1ZWYzYTc0YjMxMzIzNDFmOTIyY2FmYmU3ZTc0YmFiYWI1IiwgIjEiLCA1LCA4LCAiNTk2ZmZjMWJjMm
ZkZDFlNDM1NTYwMmYxZTRkN2JiMWI0YzRiMWJkMDk3ZWRjMjkxNGQ2NDIzM2Y1ZTQwNTc2YiJdfQ.ImU6Up_P6Oc6dRLrxUKc2YTmTRW_bWMO
J2NsqYbUvQ7kHAv30yQoOTjZlqhhk7W01cBZKvlLEEzlG0mwb9Tf4ORpH3PVoH8LRVz_b_TekkvZbq_rcyYNsNIgX7opBTN8KOOuhoan8oi9m
J09Tri3cq89fN4u1STKGSHw7jrr3rA
Component 5: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjgtODVmYTQ3NjAzODY5MWM1NTQ4YTAwYWZjYjBmZWRkMmY5NGQ2M2Y3Z
jkyMTdlNDI4NjU0NDkxNjIwMWViZGVjM2ZjMmJlY2Q4Mjg3ZjU0OWI2MTc4MGViZmU2N2U4NDFiIiwgIjEiLCA1LCA4LCAiMmNkNDRjOThkMD
c3MGIzMWViNmM5YmZmOGEwMGZiYmViZTBkOTA0NmQzYWQxNjJhNzAwMzEyYzJmMmU4YWEwNiJdfQ.X2R9Qy4dvTDlE8heVlWEYuiJxYdVDoJ1
tCGt2F_9074cboBRQoeRirx46Ox5jdV-L95aD97Z2HQz3fz2GXymIbc10tkVa880pQHkGuu1ed7YBqA7cDw1WrBIqBfbWrqna-j4A6fMYeHG_
37QX3Q6bq3f7zVsTQt3t5LEThSuU0I
Recovered secret: Agora temos um segredo extremamente confidencialroot@CSI:~/Desktop#
```

Porém, se usarmos um número superior ao quórum, por exemplo 6, também conseguimos reconstruir o segredo.

```
root@CSI:~/Desktop# python recoverSecretFromComponents-app.py 6 1 certificado.crt
Component 1: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjEtYzJiOTZmMzBlODg1ZmFkNDViZTA3ZWU5YjNkYTAwY2MwOTBkNDdhN
zRiY2ZmN2YxOTJkZThiYjk0MjcxODk0MDU0ZjYwMDM3NDkwM2IwYzcyNjQzY2Y5NTRlYzc3YThjIiwgIjEiLCA1LCA4LCAiMWU5YmIwYTUwYj
NkM2EyYmRkMWZiNWNhOTljZGVmNjAzYzU2Y2I2NDdjMzI0YzdkNDFlNzIwMmQ1NGI5NjNkNiJdfQ.R1U4w-wHlU86rfU9MlPfPBhBZG1i-PRO
f6WIsdEsqHzZ1XmUg08odzgfXRwnYAW_yc9-cYOXPSara05VU3O6FOAritJnQYKOssfWZIlC_o5doWt3cBl0h8HkNKhTRwhLD-8kcXJEHKbW9
LmEBoSomC-YdL18dNLSREmemsQBy1U
Component 2: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjItMWYxOTM3OTRkMTQ3ZjBjZGMyMTNhZWZkYjZiZmExYWVjYzUzYWY2O
GFlMDA3YmEwZGU3MTkzMDMwODlmNzVjOWEwNzU4ZWFiZTJjMTExMjljNzFiNDkxNTg0OGE2ZjQ0IiwgIjEiLCA1LCA4LCAiYTlhZGY4NDQyMD
Q2MjBkZGY3ODAyMTA1NGMyYjAzYmRkMDAyNTAzYzdmYzg0ZWI1NjZhMTllNjAwNGZkY2E0OCJdfQ.LDCZ_XhjZOS7riS66Iygu3EYeAdxR1zj
9Ib57p5ESDr935ITivf8xOTpRRq5VLKUGrWnQ4HiRySpzK0xYDgB9raTvw1ENt8WtulmcF3k-0NzUdUhjVeaYch-Swpbs1Ac8CnOWa7rM6JY0
457B0KoJ_mBdie9Nydyn1SmKoPdZmY
Component 3: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjMtYjFlY2I3NDNkMjVhNzA3OTg4YTE5YmZhYzEyYjI3OGIxMGI1MjcxN
DIwYjllNTI1MjJhZTg5MzNhOWZhYmFkYjI0NWM0ZDQyY2UyMzAyODlmYzljNWVkY2YzMWNlNGI4IiwgIjEiLCA1LCA4LCAiMzlhNGE0ZDIzYW
UwNWE5MzNiYTg2YjkyM2NkY2M3OTI0NDlkYTg5OWFjZDhlZjJiN2RhM2JlZTdhZDlmNzExMSJdfQ.d2dgbPqdqGdbyqfavwL8QIRVdui9Q8mn
t-mn02sAQD799RqzWL_4pKSQH9bcDVYlWo3uZZ0cWpkqIH9Y43tOH--vFKY53iHIEHZ0fE3myLnbMYy1SW7PcySEHFc87UOghR20zhuMkQ8QV
ujOO4tmTyCKE8MkXn9rJ_wjTbSn7bk
Component 4: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjQtMTVjNjgwNWVhNjIxODAwMWM1OTQ2NTU2MDdmZTIzNmI5ZjIwY2Y3O
WQ2MTlhMzgyZjdkNzg5MzAzZGJjN2RhZDEzNWZjYzA4ZGQ3NTI4ZGJkYmY5MTI0ODRkZThiMTBlIiwgIjEiLCA1LCA4LCAiMWY1MGI4MjgyYW
EyMDgwYWE1ZDI1YTViZDFjYTdlNTQ1YjhkZTRmZmJlZTMzMzg4YTJhYTQwY2I3YjkyZDFkNSJdfQ.fNdfXVcuhsOt24tH9se0OR35gkxFT7tf
ezwA8gwSmni86aI3S6dRK9LnllLSZHp7OfWhHgyzkj_gUWNJ4mq12w2wlwlwBQUXs43UligiBp8Ia5bam82lk2cJw-UzXbmJPuya2FLt8QmIr
qvdWXrJ3_olBIDIwuZJQ-qgBbBWktk
Component 5: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjUtMjQ2NWM4ODEwYTcxMTE5MmJjNjdiMTViYjFlZTkyZjZhN2ZhNmNjZ
jljYTE1YjExYzZlZmM3ZDIwODQ3ODBkNWZmODFlZTUzMTdkMzU5MGJkZWUzZDkyMzBkNTZlMzhkIiwgIjEiLCA1LCA4LCAiZDYyYjI1MDc4OT
g1ODU0YjFjZmVhMGE3MGI2OTY0ZjM3Yjk1NWJjYTM5MmQwMzcwZTQ2MTE5NWNjZjYwMzgxOCJdfQ.K0zBft1Gtmbdls-Wd3ggRPW4UHueDWlT
20_JPgfCGKQjJqTsRJLNYHP-aLPiz_dGIIcF6naKfddj43ZiBdqqY19r_F9qDUNF3g7TF6p-6HjmCrF7jiu9j5v-6vS0kYtBr3pgJ6f7siOuH
_Zhf8Y1Byc4lGECOyBPhWQ9IuNDHLw
Component 6: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjctZTRkMGRkYjI4ZDI5MWY4YWM4ZDYwY2M5ODMyNWUyYWRlZmNkYzgxZ
WE1MjQ1MWJlMGFjMDU2MmYzNTE0NjU1ZWYzYTc0YjMxMzIzNDFmOTIyY2FmYmU3ZTc0YmFiYWI1IiwgIjEiLCA1LCA4LCAiNTk2ZmZjMWJjMm
ZkZDFlNDM1NTYwMmYxZTRkN2JiMWI0YzRiMWJkMDk3ZWRjMjkxNGQ2NDIzM2Y1ZTQwNTc2YiJdfQ.ImU6Up_P6Oc6dRLrxUKc2YTmTRW_bWMO
J2NsqYbUvQ7kHAv30yQoOTjZlqhhk7W01cBZKvlLEEzlG0mwb9Tf4ORpH3PVoH8LRVz_b_TekkvZbq_rcyYNsNIgX7opBTN8KOOuhoan8oi9m
J09Tri3cq89fN4u1STKGSHw7jrr3rA
Recovered secret: Agora temos um segredo extremamente confidencialroot@CSI:~/Desktop#
```

No caso do ***recoverSecretFromAllComponents-app.py*** é necessário o uso de todas as componentes. Isto é, se pusermos o quórum=5 não iremos conseguir reconstruir o segredo. É necessário que o quorum seja 8, isto é, as componentes todas são necessárias.

```
root@CSI:~/Desktop# python recoverSecretFromAllComponents-app.py 5 1 certificado.crt
Component 1: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjItMWYxOTM3OTRkMTQ3ZjBjZGMyMTNhZWZkYjZiZmExYWVjYzUzYWY2O
GFlMDA3YmEwZGU3MTkzMDMwODlmNzVjOWEwNzU4ZWFiZTJjMTExMjljNzFiNDkxNTg0OGE2ZjQ0IiwgIjEiLCA1LCA4LCAiYTlhZGY4NDQyMD
Q2MjBkZGY3ODAyMTA1NGMyYjAzYmRkMDAyNTAzYzdmYzg0ZWI1NjZhMTllNjAwNGZkY2E0OCJdfQ.LDCZ_XhjZOS7riS66Iygu3EYeAdxR1zj
9Ib57p5ESDr935ITivf8xOTpRRq5VLKUGrWnQ4HiRySpzK0xYDgB9raTvw1ENt8WtulmcF3k-0NzUdUhjVeaYch-Swpbs1Ac8CnOWa7rM6JY0
457B0KoJ_mBdie9Nydyn1SmKoPdZmY
Component 2: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjMtYjFlY2I3NDNkMjVhNzA3OTg4YTE5YmZhYzEyYjI3OGIxMGI1MjcxN
DIwYjllNTI1MjJhZTg5MzNhOWZhYmFkYjI0NWM0ZDQyY2UyMzAyODlmYzljNWVkY2YzMWNlNGI4IiwgIjEiLCA1LCA4LCAiMzlhNGE0ZDIzYW
UwNWE5MzNiYTg2YjkyM2NkY2M3OTI0NDlkYTg5OWFjZDhlZjJiN2RhM2JlZTdhZDlmNzExMSJdfQ.d2dgbPqdqGdbyqfavwL8QIRVdui9Q8mn
t-mn02sAQD799RqzWL_4pKSQH9bcDVYlWo3uZZ0cWpkqIH9Y43tOH--vFKY53iHIEHZ0fE3myLnbMYy1SW7PcySEHFc87UOghR20zhuMkQ8QV
ujOO4tmTyCKE8MkXn9rJ_wjTbSn7bk
Component 3: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjQtMTVjNjgwNWVhNjIxODAwMWM1OTQ2NTU2MDdmZTIzNmI5ZjIwY2Y3O
WQ2MTlhMzgyZjdkNzg5MzAzZGJjN2RhZDEzNWZjYzA4ZGQ3NTI4ZGJkYmY5MTI0ODRkZThiMTBlIiwgIjEiLCA1LCA4LCAiMWY1MGI4MjgyYW
EyMDgwYWE1ZDI1YTViZDFjYTdlNTQ1YjhkZTRmZmJlZTMzMzg4YTJhYTQwY2I3YjkyZDFkNSJdfQ.fNdfXVcuhsOt24tH9se0OR35gkxFT7tf
ezwA8gwSmni86aI3S6dRK9LnllLSZHp7OfWhHgyzkj_gUWNJ4mq12w2wlwlwBQUXs43UligiBp8Ia5bam82lk2cJw-UzXbmJPuya2FLt8QmIr
qvdWXrJ3_olBIDIwuZJQ-qgBbBWktk
Component 4: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjUtMjQ2NWM4ODEwYTcxMTE5MmJjNjdiMTViYjFlZTkyZjZhN2ZhNmNjZ
jljYTE1YjExYzZlZmM3ZDIwODQ3ODBkNWZmODFlZTUzMTdkMzU5MGJkZWUzZDkyMzBkNTZlMzhkIiwgIjEiLCA1LCA4LCAiZDYyYjI1MDc4OT
g1ODU0YjFjZmVhMGE3MGI2OTY0ZjM3Yjk1NWJjYTM5MmQwMzcwZTQ2MTE5NWNjZjYwMzgxOCJdfQ.K0zBft1Gtmbdls-Wd3ggRPW4UHueDWlT
20_JPgfCGKQjJqTsRJLNYHP-aLPiz_dGIIcF6naKfddj43ZiBdqqY19r_F9qDUNF3g7TF6p-6HjmCrF7jiu9j5v-6vS0kYtBr3pgJ6f7siOuH
_Zhf8Y1Byc4lGECOyBPhWQ9IuNDHLw
Component 5: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjctZTRkMGRkYjI4ZDI5MWY4YWM4ZDYwY2M5ODMyNWUyYWRlZmNkYzgxZ
WE1MjQ1MWJlMGFjMDU2MmYzNTE0NjU1ZWYzYTc0YjMxMzIzNDFmOTIyY2FmYmU3ZTc0YmFiYWI1IiwgIjEiLCA1LCA4LCAiNTk2ZmZjMWJjMm
ZkZDFlNDM1NTYwMmYxZTRkN2JiMWI0YzRiMWJkMDk3ZWRjMjkxNGQ2NDIzM2Y1ZTQwNTc2YiJdfQ.ImU6Up_P6Oc6dRLrxUKc2YTmTRW_bWMO
J2NsqYbUvQ7kHAv30yQoOTjZlqhhk7W01cBZKvlLEEzlG0mwb9Tf4ORpH3PVoH8LRVz_b_TekkvZbq_rcyYNsNIgX7opBTN8KOOuhoan8oi9m
J09Tri3cq89fN4u1STKGSHw7jrr3rA
Error: Invalid number of componentsroot@CSI:~/Desktop#
```

```
root@CSI:~/Desktop# python recoverSecretFromAllComponents-app.py 8 1 certificado.crt
Component 1: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjEtYzJiOTZmMzBlODg1ZmFkNDViZTA3ZWU5YjNkYTAwY2MwOTBkNDdhN
zRiY2ZmN2YxOTJkZThiYjk0MjcxODk0MDU0ZjYwMDM3NDkwM2IwYzcyNjQzY2Y5NTRlYzc3YThjIiwgIjEiLCA1LCA4LCAiMWU5YmIwYTUwYj
NkM2EyYmRkMWZiNWNhOTljZGVmNjAzYzU2Y2I2NDdjMzI0YzdkNDFlNzIwMmQ1NGI5NjNkNiJdfQ.R1U4w-wHlU86rfU9MlPfPBhBZG1i-PRO
f6WIsdEsqHzZ1XmUg08odzgfXRwnYAW_yc9-cYOXPSara05VU3O6FOAritJnQYKOssfWZIlC_o5doWt3cBl0h8HkNKhTRwhLD-8kcXJEHKbW9
LmEBoSomC-YdL18dNLSREmemsQBy1U
Component 2: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjItMWYxOTM3OTRkMTQ3ZjBjZGMyMTNhZWZkYjZiZmExYWVjYzUzYWY2O
GFlMDA3YmEwZGU3MTkzMDMwODlmNzVjOWEwNzU4ZWFiZTJjMTExMjljNzFiNDkxNTg0OGE2ZjQ0IiwgIjEiLCA1LCA4LCAiYTlhZGY4NDQyMD
Q2MjBkZGY3ODAyMTA1NGMyYjAzYmRkMDAyNTAzYzdmYzg0ZWI1NjZhMTllNjAwNGZkY2E0OCJdfQ.LDCZ_XhjZOS7riS66Iygu3EYeAdxR1zj
9Ib57p5ESDr935ITivf8xOTpRRq5VLKUGrWnQ4HiRySpzK0xYDgB9raTvw1ENt8WtulmcF3k-0NzUdUhjVeaYch-Swpbs1Ac8CnOWa7rM6JY0
457B0KoJ_mBdie9Nydyn1SmKoPdZmY
Component 3: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjMtYjFlY2I3NDNkMjVhNzA3OTg4YTE5YmZhYzEyYjI3OGIxMGI1MjcxN
DIwYjllNTI1MjJhZTg5MzNhOWZhYmFkYjI0NWM0ZDQyY2UyMzAyODlmYzljNWVkY2YzMWNlNGI4IiwgIjEiLCA1LCA4LCAiMzlhNGE0ZDIzYW
UwNWE5MzNiYTg2YjkyM2NkY2M3OTI0NDlkYTg5OWFjZDhlZjJiN2RhM2JlZTdhZDlmNzExMSJdfQ.d2dgbPqdqGdbyqfavwL8QIRVdui9Q8mn
t-mn02sAQD799RqzWL_4pKSQH9bcDVYlWo3uZZ0cWpkqIH9Y43tOH--vFKY53iHIEHZ0fE3myLnbMYy1SW7PcySEHFc87UOghR20zhuMkQ8QV
ujOO4tmTyCKE8MkXn9rJ_wjTbSn7bk
Component 4: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjQtMTVjNjgwNWVhNjIxODAwMWM1OTQ2NTU2MDdmZTIzNmI5ZjIwY2Y3O
WQ2MTlhMzgyZjdkNzg5MzAzZGJjN2RhZDEzNWZjYzA4ZGQ3NTI4ZGJkYmY5MTI0ODRkZThiMTBlIiwgIjEiLCA1LCA4LCAiMWY1MGI4MjgyYW
EyMDgwYWE1ZDI1YTViZDFjYTdlNTQ1YjhkZTRmZmJlZTMzMzg4YTJhYTQwY2I3YjkyZDFkNSJdfQ.fNdfXVcuhsOt24tH9se0OR35gkxFT7tf
ezwA8gwSmni86aI3S6dRK9LnllLSZHp7OfWhHgyzkj_gUWNJ4mq12w2wlwlwBQUXs43UligiBp8Ia5bam82lk2cJw-UzXbmJPuya2FLt8QmIr
qvdWXrJ3_olBIDIwuZJQ-qgBbBWktk
Component 5: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjUtMjQ2NWM4ODEwYTcxMTE5MmJjNjdiMTViYjFlZTkyZjZhN2ZhNmNjZ
jljYTE1YjExYzZlZmM3ZDIwODQ3ODBkNWZmODFlZTUzMTdkMzU5MGJkZWUzZDkyMzBkNTZlMzhkIiwgIjEiLCA1LCA4LCAiZDYyYjI1MDc4OT
g1ODU0YjFjZmVhMGE3MGI2OTY0ZjM3Yjk1NWJjYTM5MmQwMzcwZTQ2MTE5NWNjZjYwMzgxOCJdfQ.K0zBft1Gtmbdls-Wd3ggRPW4UHueDWlT
20_JPgfCGKQjJqTsRJLNYHP-aLPiz_dGIIcF6naKfddj43ZiBdqqY19r_F9qDUNF3g7TF6p-6HjmCrF7jiu9j5v-6vS0kYtBr3pgJ6f7siOuH
_Zhf8Y1Byc4lGECOyBPhWQ9IuNDHLw
Component 6: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjYtZjZiNjY4YzFjMDhkMDM1YWRlMDhhZDJlZDc4NmUwNmRjMzFhNjZiM
GRmMzZlNTdiYzliYjkzZTY3YjI4MjQ0YWRhMTBlYWJlYjkyOTk4ZmVlMzhmOWRhNzZlY2ZiZjMzIiwgIjEiLCA1LCA4LCAiM2ZjOTgxOWNmYz
BkOTI0ZTkzYmVmZmI4Y2M4MjlkNmRjMjliNDZmZTc5MjI3ZWM4M2I2NGQ1YThmYzI2ZWYxOCJdfQ.A_PcHjN-wPGUCVQ_KpuUfW_pzCSq9Pch
OlSMyxkR5QJtD9b20bbVCt_bOHJ5seX-J-XOVwDHedW6padbYcE6VLa4QJmXix55VsmSmqhrbr8va5EKmvWZk0fFLBYVx1igWcTfVeF1iJ0cs
egxA4KcIi7K9EJJof6-KB6gybENwto
Component 7: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjctZTRkMGRkYjI4ZDI5MWY4YWM4ZDYwY2M5ODMyNWUyYWRlZmNkYzgxZ
WE1MjQ1MWJlMGFjMDU2MmYzNTE0NjU1ZWYzYTc0YjMxMzIzNDFmOTIyY2FmYmU3ZTc0YmFiYWI1IiwgIjEiLCA1LCA4LCAiNTk2ZmZjMWJjMm
ZkZDFlNDM1NTYwMmYxZTRkN2JiMWI0YzRiMWJkMDk3ZWRjMjkxNGQ2NDIzM2Y1ZTQwNTc2YiJdfQ.ImU6Up_P6Oc6dRLrxUKc2YTmTRW_bWMO
J2NsqYbUvQ7kHAv30yQoOTjZlqhhk7W01cBZKvlLEEzlG0mwb9Tf4ORpH3PVoH8LRVz_b_TekkvZbq_rcyYNsNIgX7opBTN8KOOuhoan8oi9m
J09Tri3cq89fN4u1STKGSHw7jrr3rA
Component 8: eyJhbGciOiAiUlMyNTYifQ.eyJvYmplY3QiOiBbIjgtODVmYTQ3NjAzODY5MWM1NTQ4YTAwYWZjYjBmZWRkMmY5NGQ2M2Y3Z
jkyMTdlNDI4NjU0NDkxNjIwMWViZGVjM2ZjMmJlY2Q4Mjg3ZjU0OWI2MTc4MGViZmU2N2U4NDFiIiwgIjEiLCA1LCA4LCAiMmNkNDRjOThkMD
c3MGIzMWViNmM5YmZmOGEwMGZiYmViZTBkOTA0NmQzYWQxNjJhNzAwMzEyYzJmMmU4YWEwNiJdfQ.X2R9Qy4dvTDlE8heVlWEYuiJxYdVDoJ1
tCGt2F_9074cboBRQoeRirx46Ox5jdV-L95aD97Z2HQz3fz2GXymIbc10tkVa880pQHkGuu1ed7YBqA7cDw1WrBIqBfbWrqna-j4A6fMYeHG_
37QX3Q6bq3f7zVsTQt3t5LEThSuU0I
Recovered secret: Agora temos um segredo extremamente confidencialroot@CSI:~/Desktop#
```

Podemos concluir assim que o ***recoverSecretFromAllComponents-app.py*** serve apenas quando queremos que sejam necessárias todas as partes presentes para obter o segredo. Já no caso do ***recoverSecretFromComponents-app.py,*** podemos utilizar quando queremos que apenas N partes, do total das partes, sejam necessárias para obter o segredo.

3. Authenticated Encryption

- ***Pergunta 3.1***

A encriptação autenticada garante simultaneamente a confidencialidade e a autenticidade. Para garantir integridade da mensagem associamos um Mac. Usamos assim o modo Encrypt-Then-Mac.

Cifração:

Cifra ← Enc(mensagem, chave)
Mac ← Hash(cifra)
Return (Cifra || Mac)

Decifração:

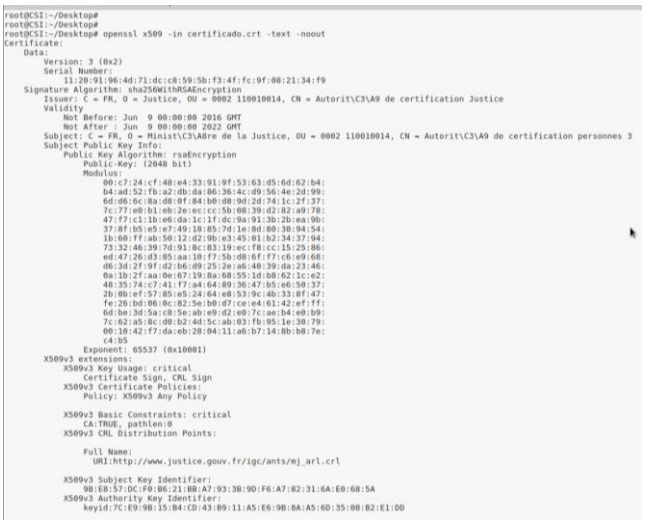
Decifra ← Dec(Cifra, chave, Mac)
Return (Decifra)

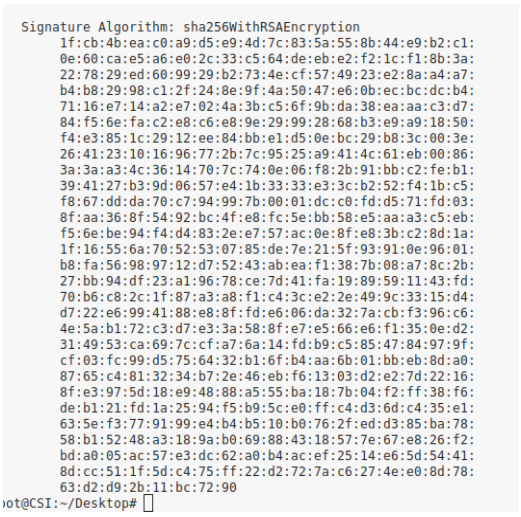
4. Algoritmos e tamanhos de chaves

- ***Pergunta 4.1***

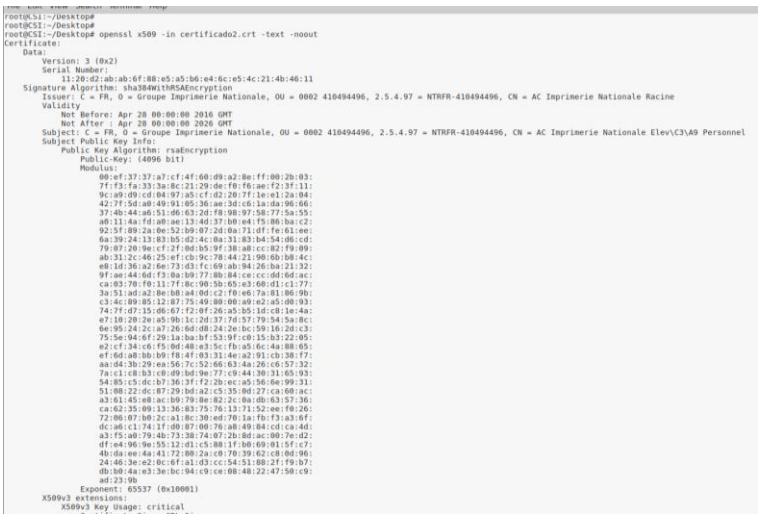
O site https://webgate.ec.europa.eu/tl-browser/ disponibiliza a lista de Entidades com serviços qualificados de confiança, de acordo com o Regulamento EU 910/2014 (eIDAS). Para a realização desta questão foram selecionadas seguintes Entidades de Certicação Francesas (EC).

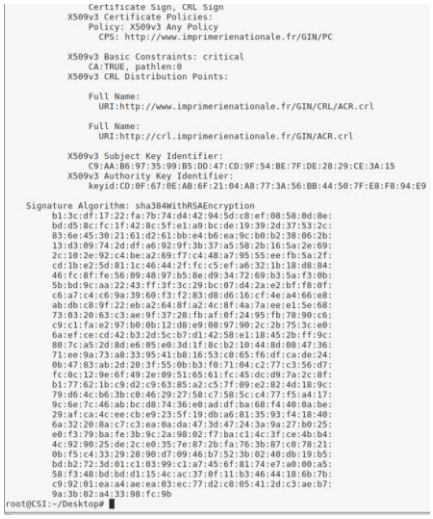
a) *Ministère de la Justice*





b) *Imprimerie Nationale*





A seguinte tabela mostra um breve resumo dos resultados obtidos através da análise das imagens acima de casa uma das Entidade de Certificação.

	Tamanho da chave	**Algoritmo**
Ministère de la Justice	2048 bits	RSA
Imprimerie Nationale	4096 bits	RSA

Assim, podemos concluir que os certificados do Imprimerie Nationale como têm uma chave de 4096 bits são mais seguros que os certificados do Ministère de la Justice que apenas possuem uma chave com 2048 bits.