



{ Reprograma }

AUTENTICAÇÃO E AUTORIZAÇÃO

CONHECENDO
JWT - JSON WEB
TOKEN



A portrait of a young woman with short dark hair, featuring a blue-to-black gradient on the right side. She has a nose piercing and is wearing a light blue denim shirt. The background is a textured wall.

Olá meu nome é Carol { Doguinho } serei sua Professora hoje!

Atualmente estudante de Análise e Desenvolvimento de Sistemas, em transição de carreira fazendo freelancer com desenvolvimento de páginas web, monitora na {Reprograma}, Jogadora de Flag Football, apaixonada por pets, animes e séries.

 doguinhoweb

 linkedin.com/in/carolalves90

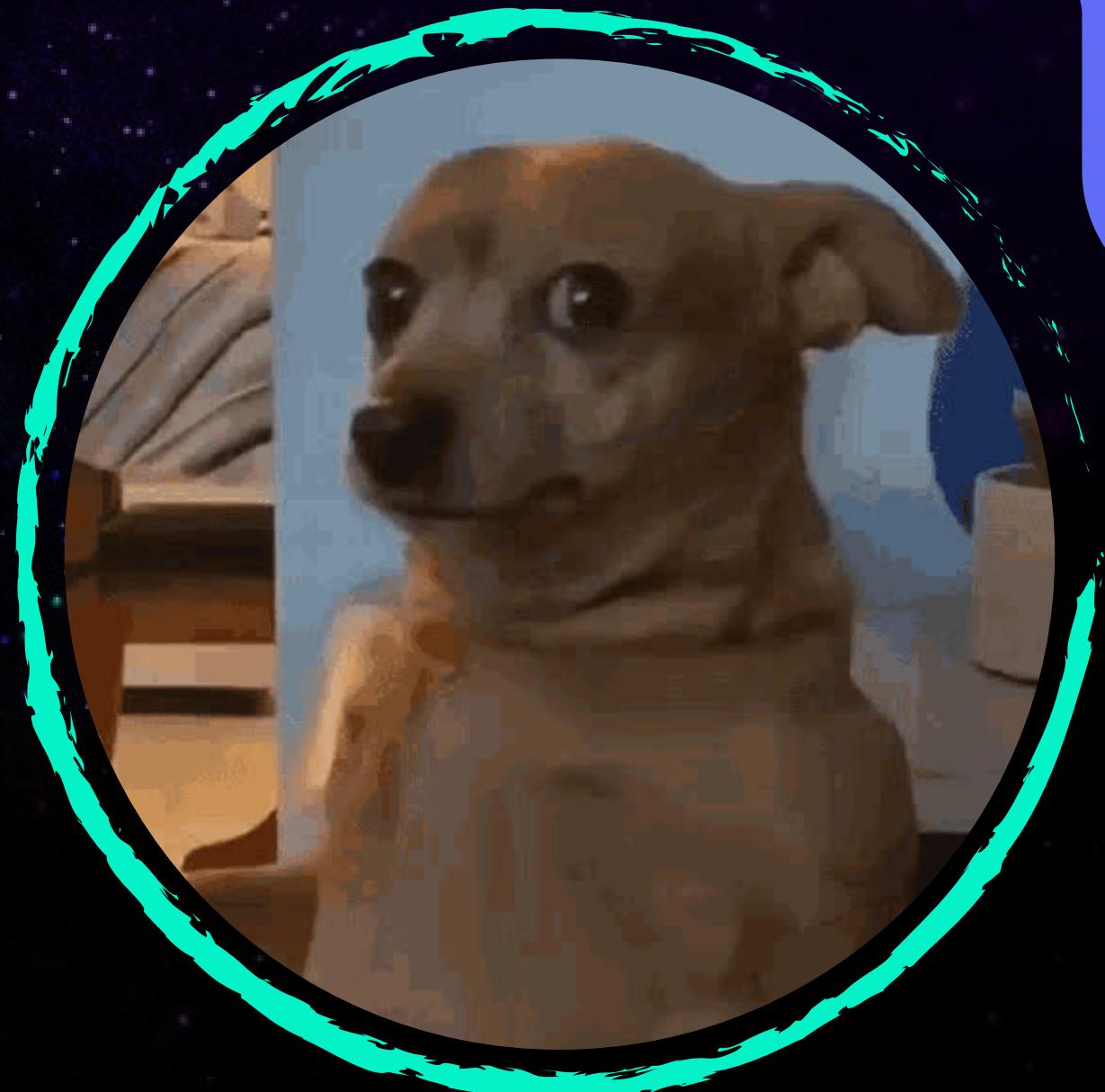
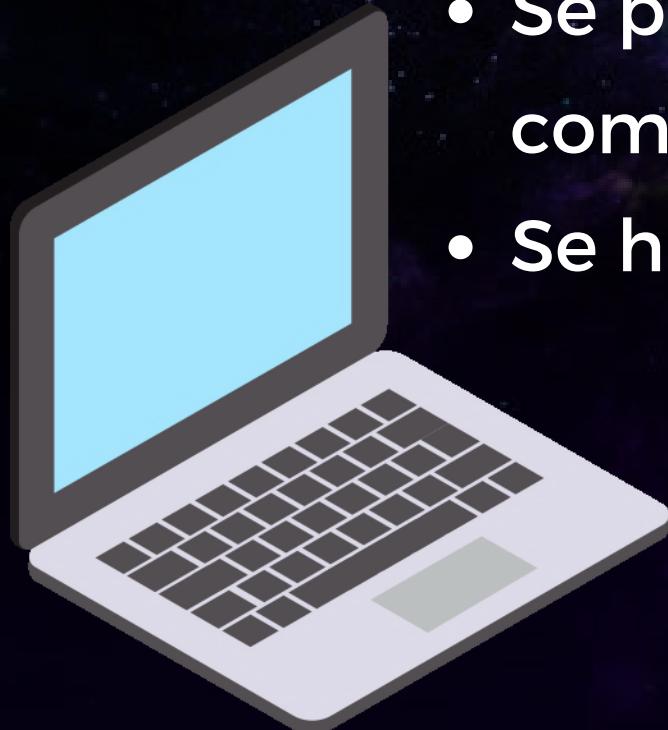
<Monitoras de hoje>



<Instruções>

Antes de começar, vamos fazer alguns combinados:

- Se tiver alguma dúvida, use o botão levantar a mão do zoom;
- As monitoras estarão de plantão caso haja dúvidas pelo chat;
- Se possível deixe a câmera ligada! E cuidado com o vazamento de áudio!
- Se hidratem, meus cactos!



O que veremos hoje?

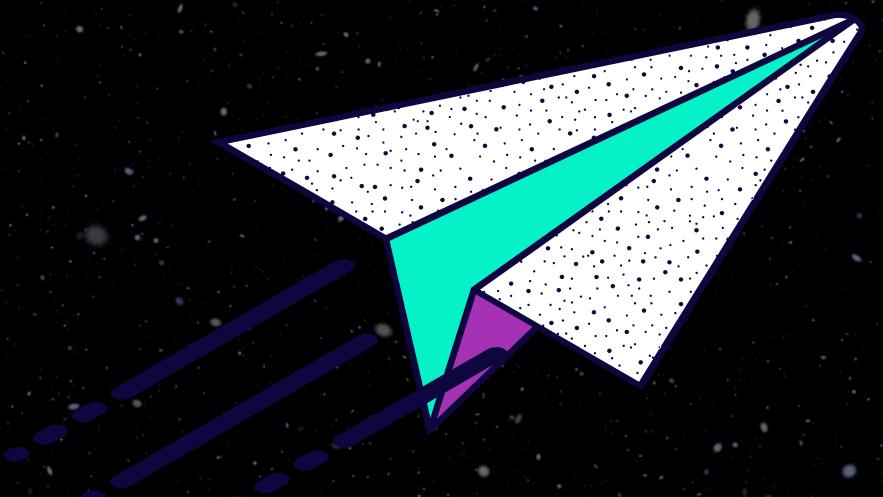
AUTENTICAÇÃO

MÉTODOS DE
AUTENTICAÇÃO

DÚVIDAS

CRPTOGRAFIA

MÃO NA MASSA



Segurança da nossa API

Nas aulas anteriores você aprendeu sobre os métodos HTTP e banco de dados.

Entretanto, qualquer pessoa que tiver acesso a sua API poderá utilizá-las livremente para alterar , salvar e deletar informações sem o menor tipo de controle?

Como podemos proteger nossas rotas?

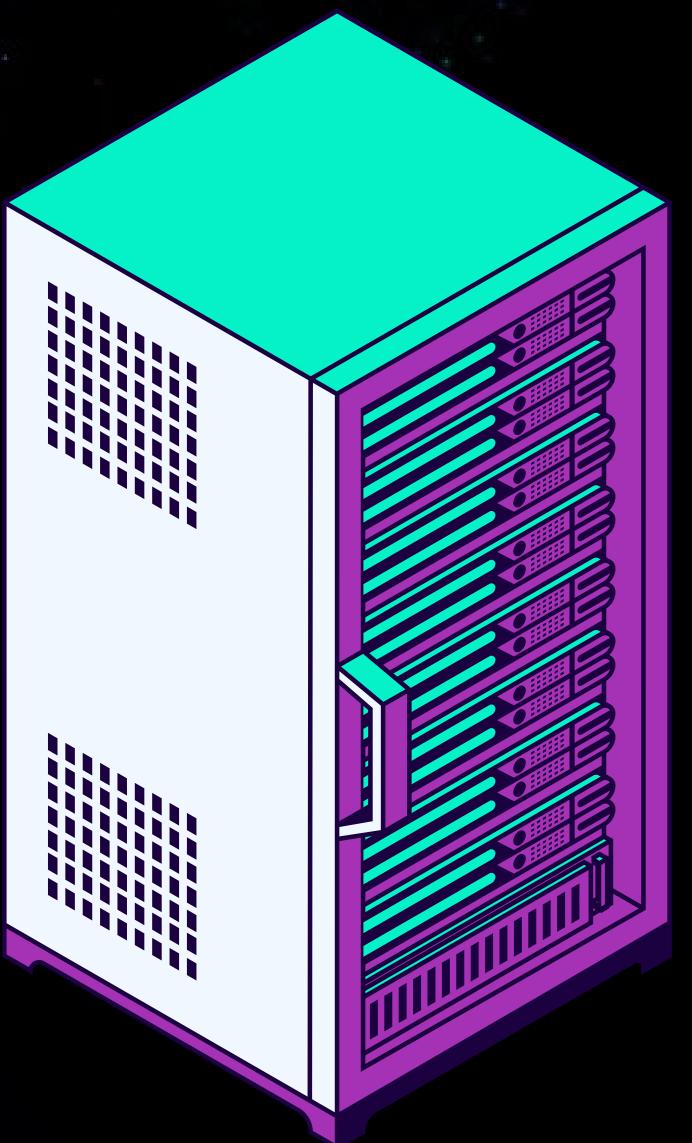
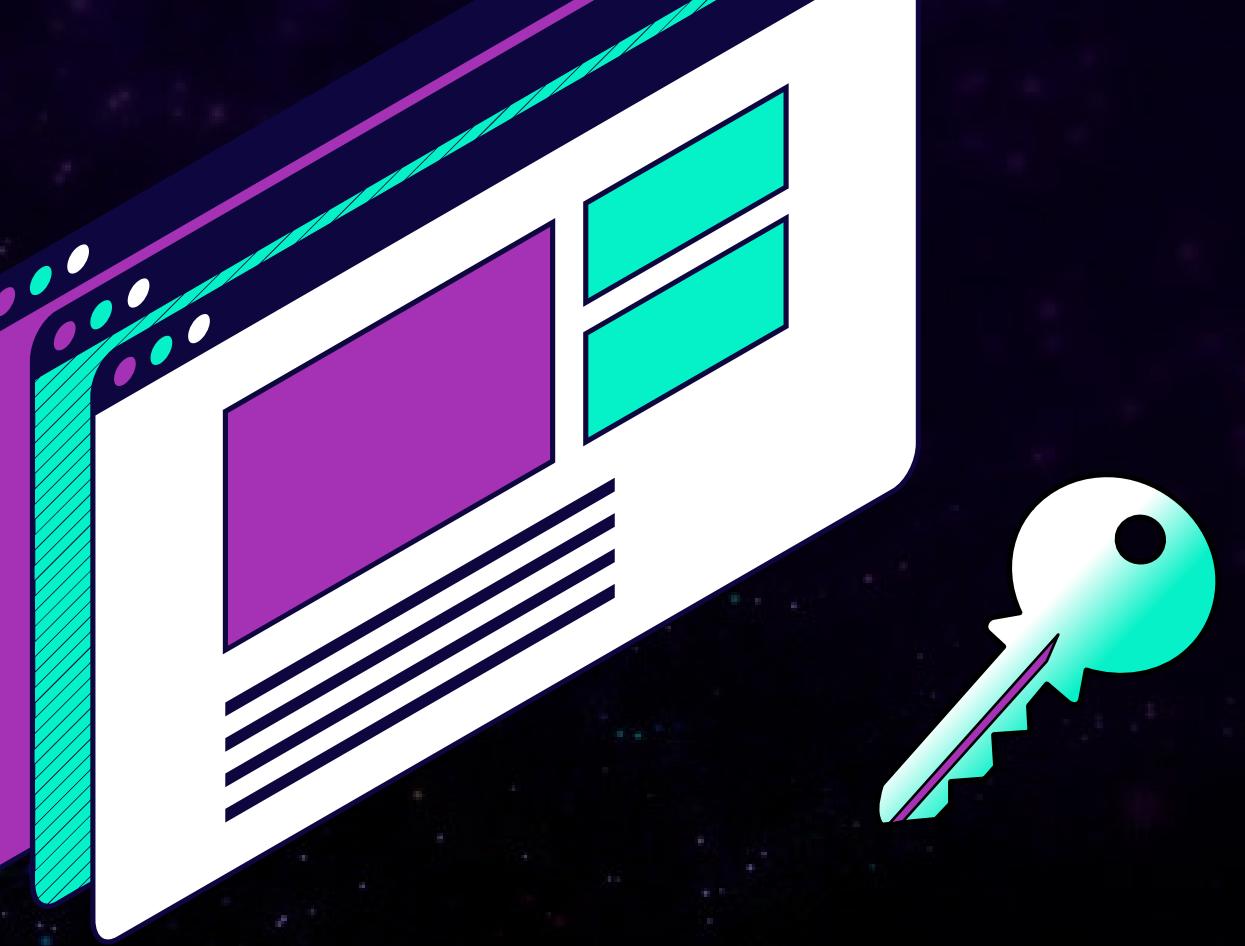
Meu amigo: Qual sua senha?
Eu: 138184288
Meu amigo: Por que essa senha?
Eu:



Aluminium 13 Al	Oxygène 8 O	Hydrogène 1 H	Oxygène 8 O	Molybdène 42 Mo	Radium 88 Ra
26,9815385	15,99940	1,007975	15,99940	95,95 (1)	[226]

Responda:

**Qual a
diferença entre
Autenticação e
Autorização?**



AUTENTICAÇÃO X AUTORIZAÇÃO

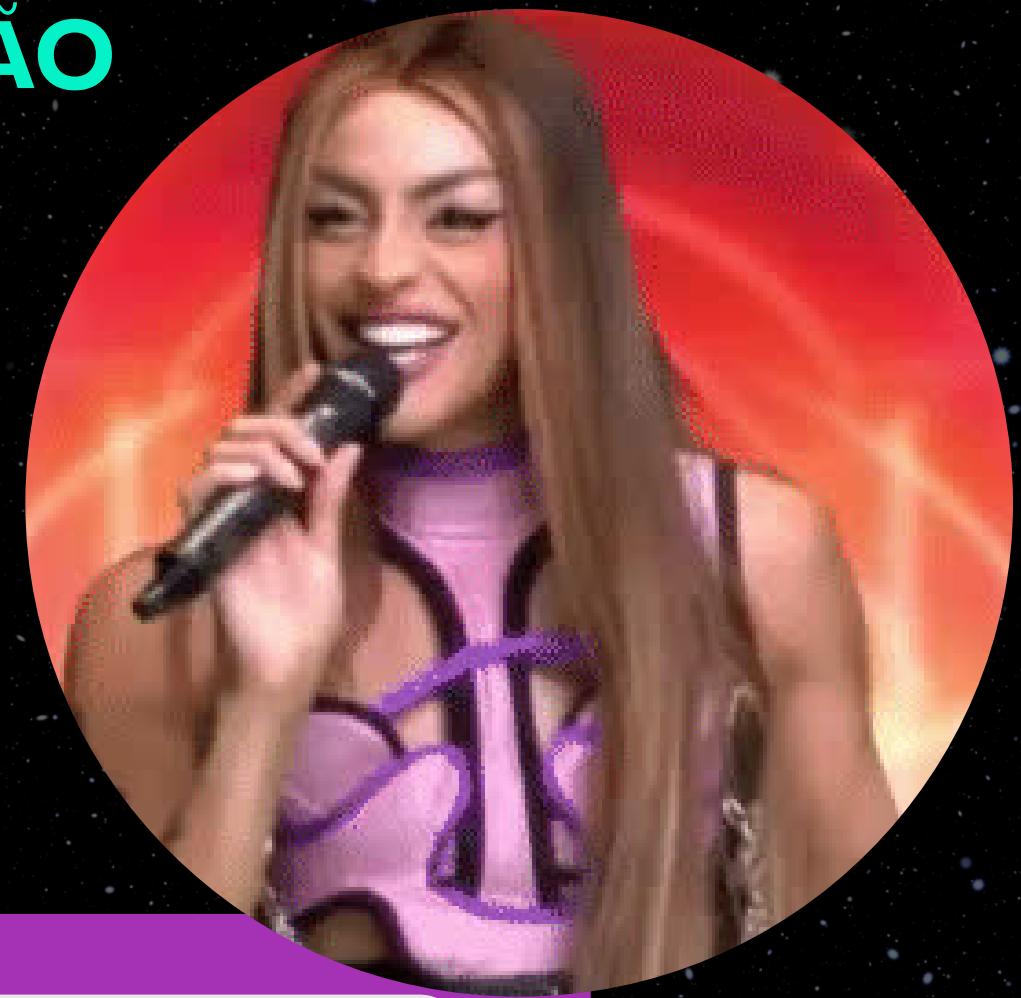
Autenticação:

Ação de efetuar o login, passando nome, email, senha e o servidor devolver a resposta.

Autorização:

A partir da autenticação o servidor irá responder se está autorizado a acessar o sistema e qual tipo de acesso.

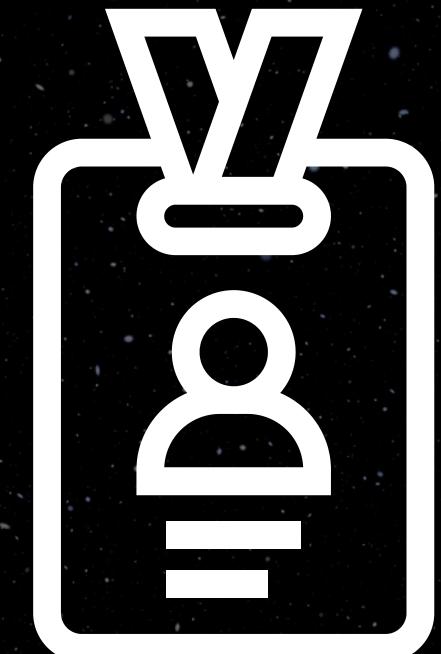
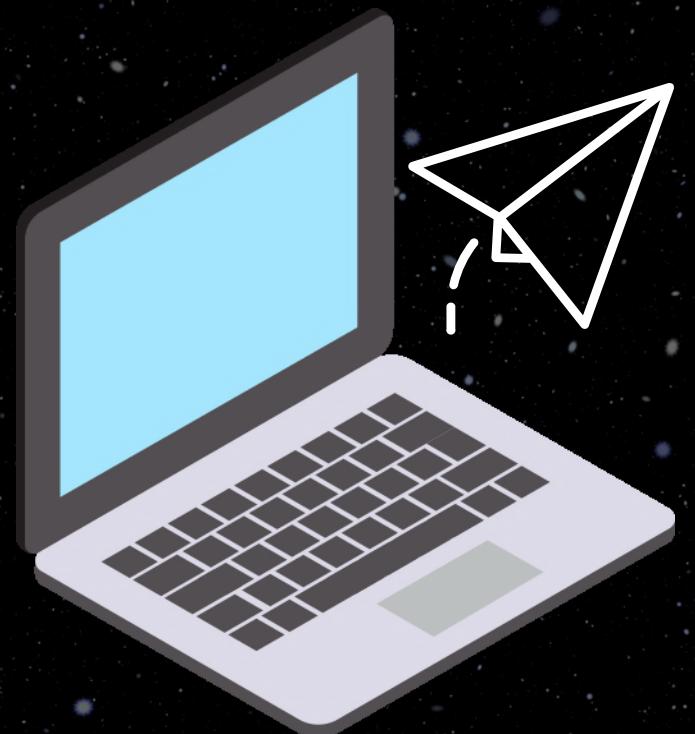
Checagem de acesso.



Remember me

[Forgot my password](#)

EXEMPLIC



ACESSO
PISTA VIP
NOVEMBER 2023
31.11.23



**Apresentei meu
Ingresso**



**Passei pela segurança que
autorizou minha entrada**

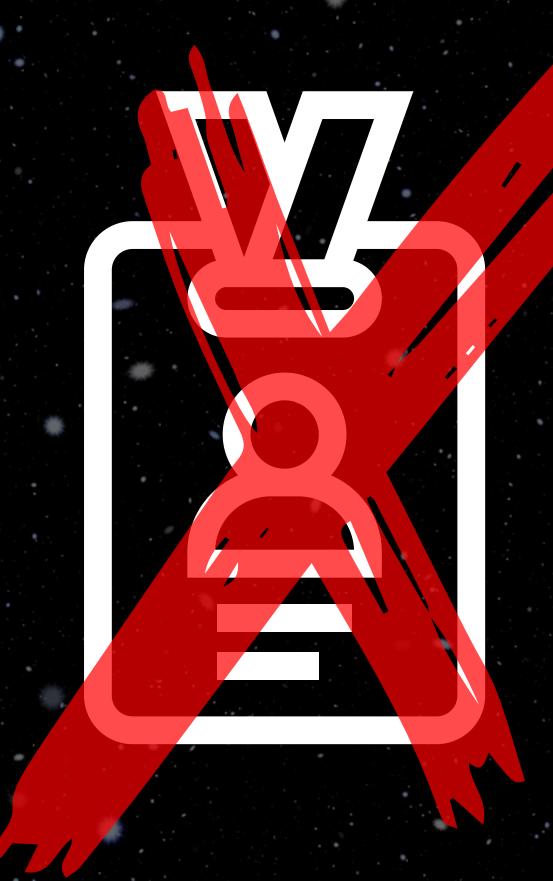
Assisti ao show...

E depois de 1h30 de show



Sai correndo pro camarim

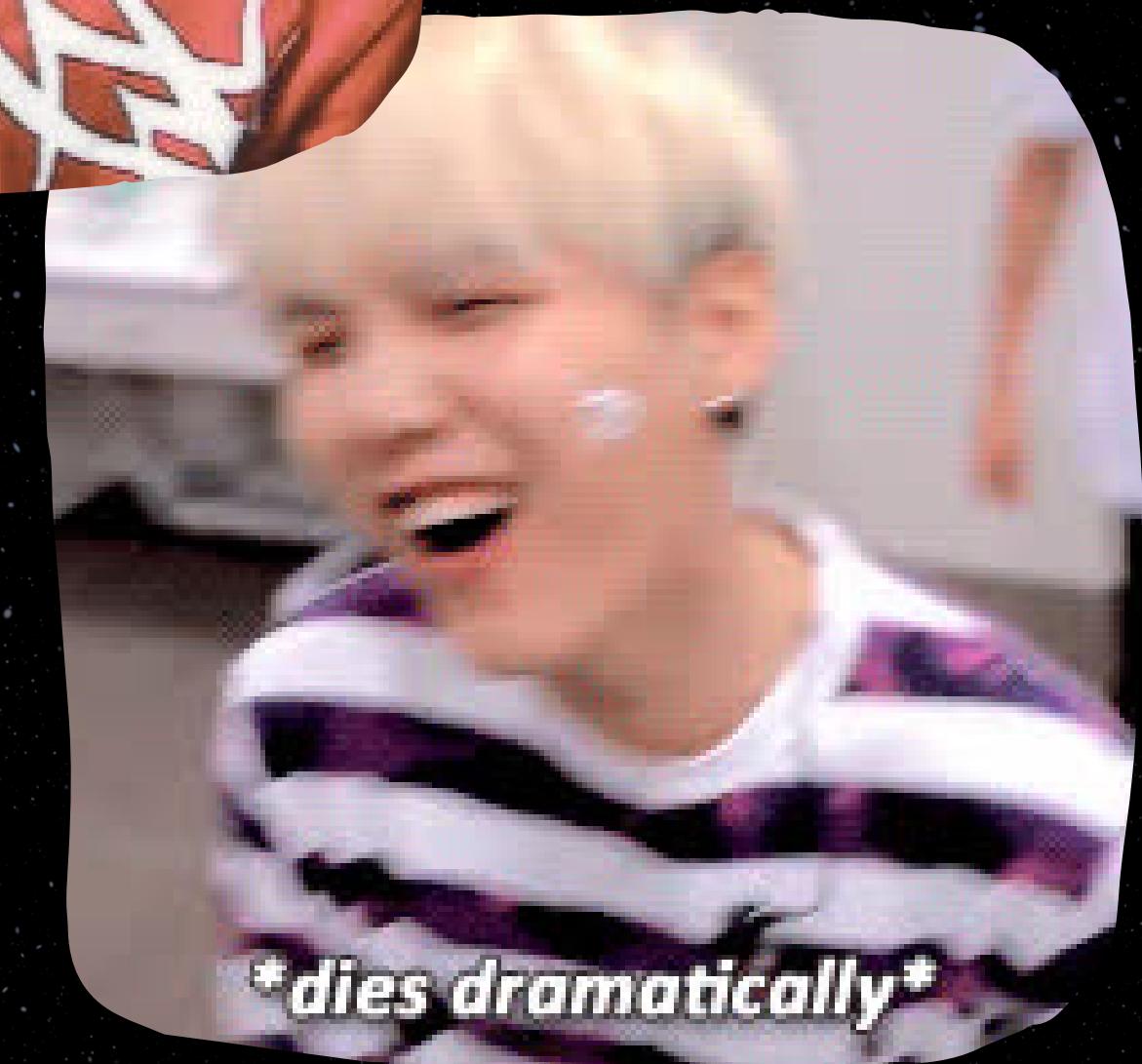
Só que chegando lá...



ACESSO AO
CAMARIM

NOVEMBER 2023

31.11.23



O segurança não autoriza a entrada

dies dramatically

EIS A QUESTÃO:

No meu exemplo o que você conseguiu identificar que era uma autenticação e o que era a autorização ?

AUTENTICAÇÃO

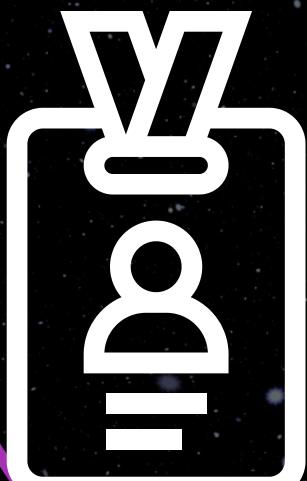


BTS BRAZILIAN
TOUR

NOVEMBER 2023

31.11.23

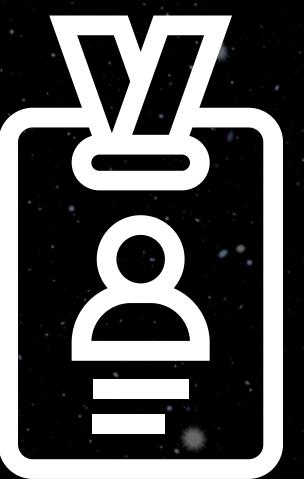
AUTORIZAÇÃO



ACESSO
PISTA VIP

NOVEMBER 2023

31.11.23



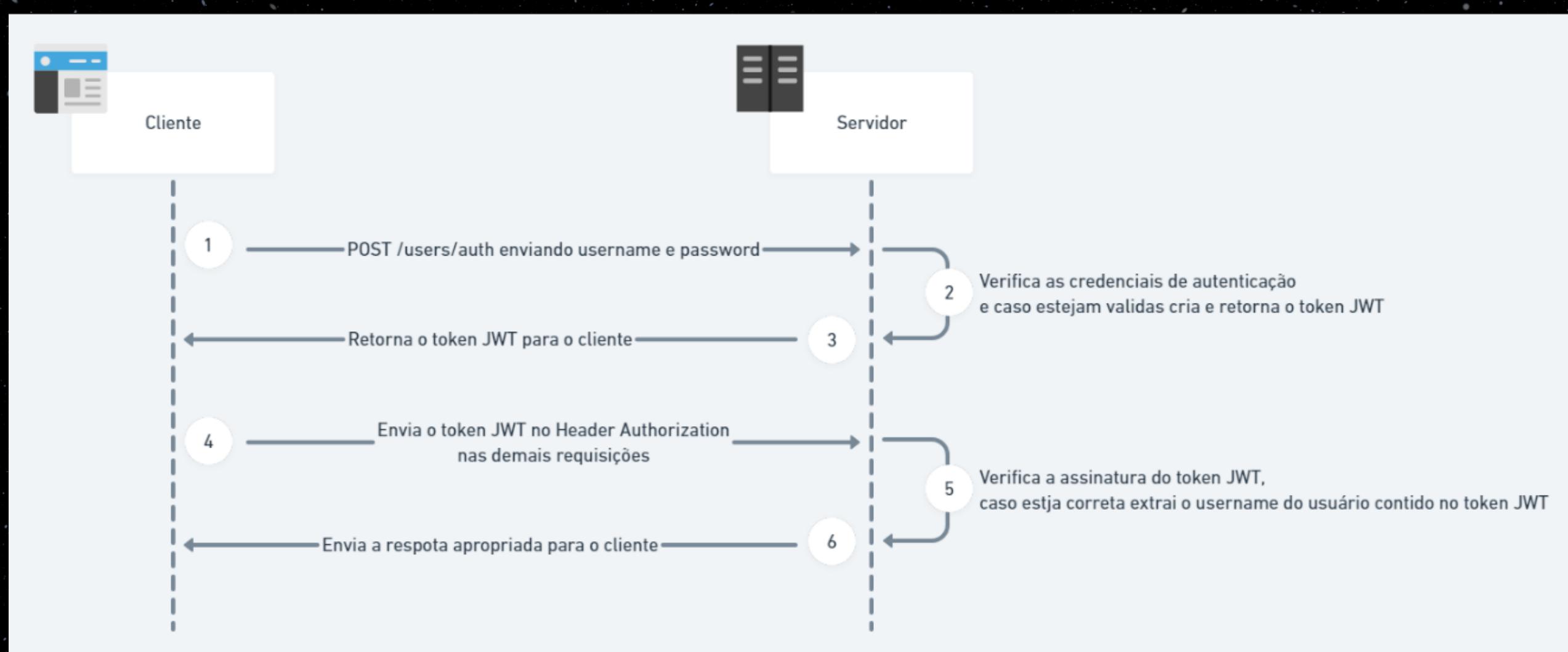
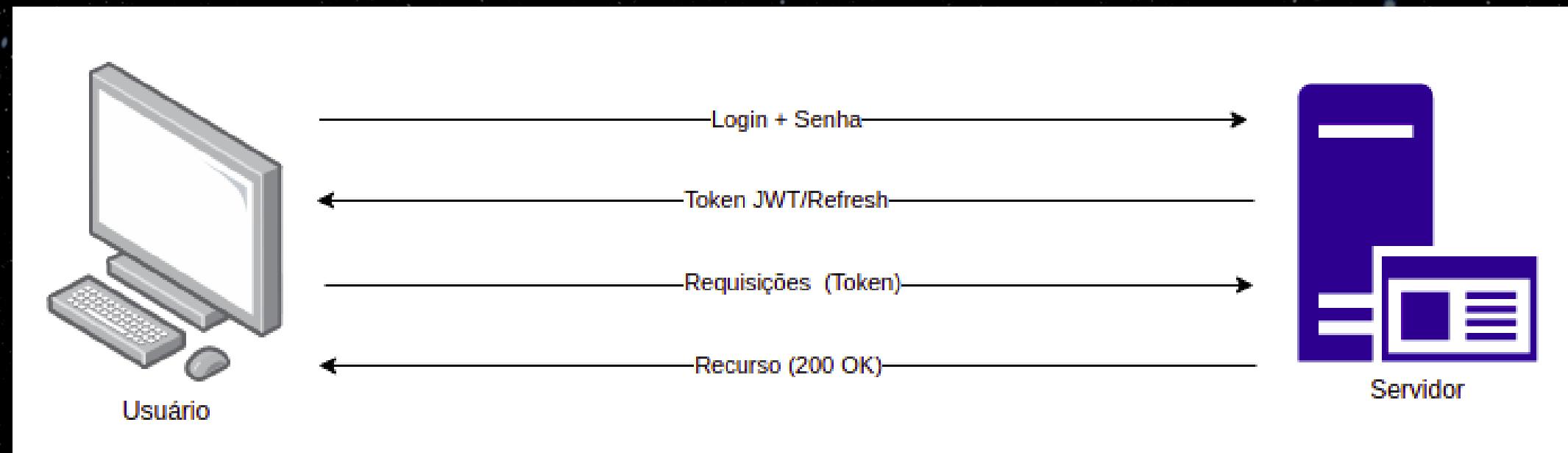
ACESSO AO
CAMARIM

NOVEMBER 2023

31.11.23

CREDENCIAIS

AUTENTICAÇÃO - COMO FUNCIONA?



NO POSTMAN

The screenshot shows a browser-based API testing interface. At the top, there's a header bar with tabs for 'Teste' and 'POST New Request'. The main title is 'HTTP Teste / New Request'. On the right side of the header are 'Save', 'Edit', and 'Delete' buttons. The status bar at the top right says 'No Environment'.

The main area is a request configuration panel. It shows a 'POST' method and the URL 'http://localhost:2323/colaboradoras/login'. Below the method and URL are tabs for 'Params', 'Authorization', 'Headers (8)', 'Body' (which is selected), 'Pre-request Script', 'Tests', and 'Settings'. The 'Body' tab has a dropdown menu with options: 'none', 'form-data', 'x-www-form-urlencoded', 'raw', 'binary', 'GraphQL', and 'JSON' (which is currently selected). To the right of the tabs are 'Cookies' and 'Beautify' buttons.

The 'Body' section contains a code editor with the following JSON payload:

```
1 {  
2   "email": "doguinhoweb@email.com",  
3   "senha": "alohomora"  
4 }
```

Below the request configuration, there are tabs for 'Body', 'Cookies', 'Headers (7)', and 'Test Results'. The 'Body' tab is selected. The status bar at the bottom shows 'Status: 200 OK Time: 114 ms Size: 377 B'. There are also 'Save as Example' and 'More' buttons.

The 'Body' section displays the response content in different formats: 'Pretty' (showing the JSON structure), 'Raw' (showing the raw JSON string), 'Preview' (showing a preview of the JSON object), and 'HTML' (showing the JSON as an HTML table). The 'Pretty' tab is selected.

The response body is a single line of JSON:

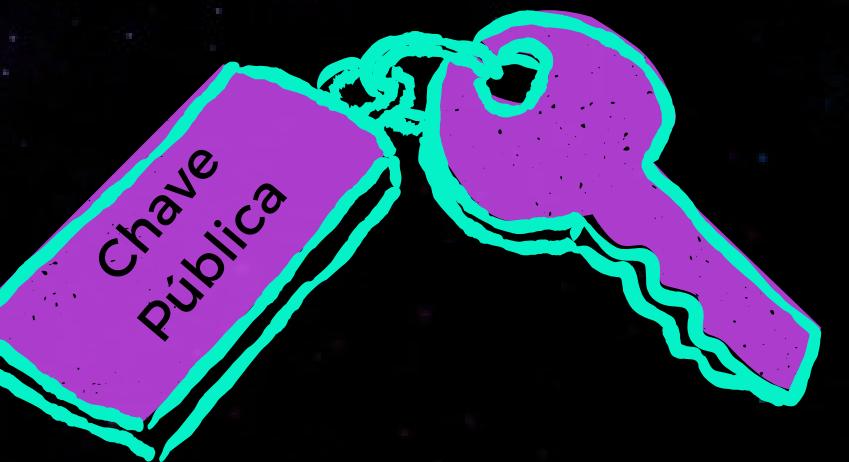
```
1 eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlbWFpbCI6ImRvZ3Vpbmhvd2ViQGVtYWlsLmNvbSIsImhlhdCI6MTY4Njk2MDg5N30.  
FRe3TYSK9ljqyDE3wcWajX9x4YxH-dkWHzUdM9TVwSk4
```

At the bottom of the interface are buttons for 'Runner', 'Capture requests', 'Cookies', 'Trash', and a 'More' button.



CRIPTOGRAFIA

Chave privada



{ **** }

CRIPTOGRAFIA

Criptografar é codificar uma mensagem que você não quer que pessoas não autorizadas tenham acesso.

Para armazenar nossas senhas na base de dados, podemos utilizar um código hash



Dica de filmes:

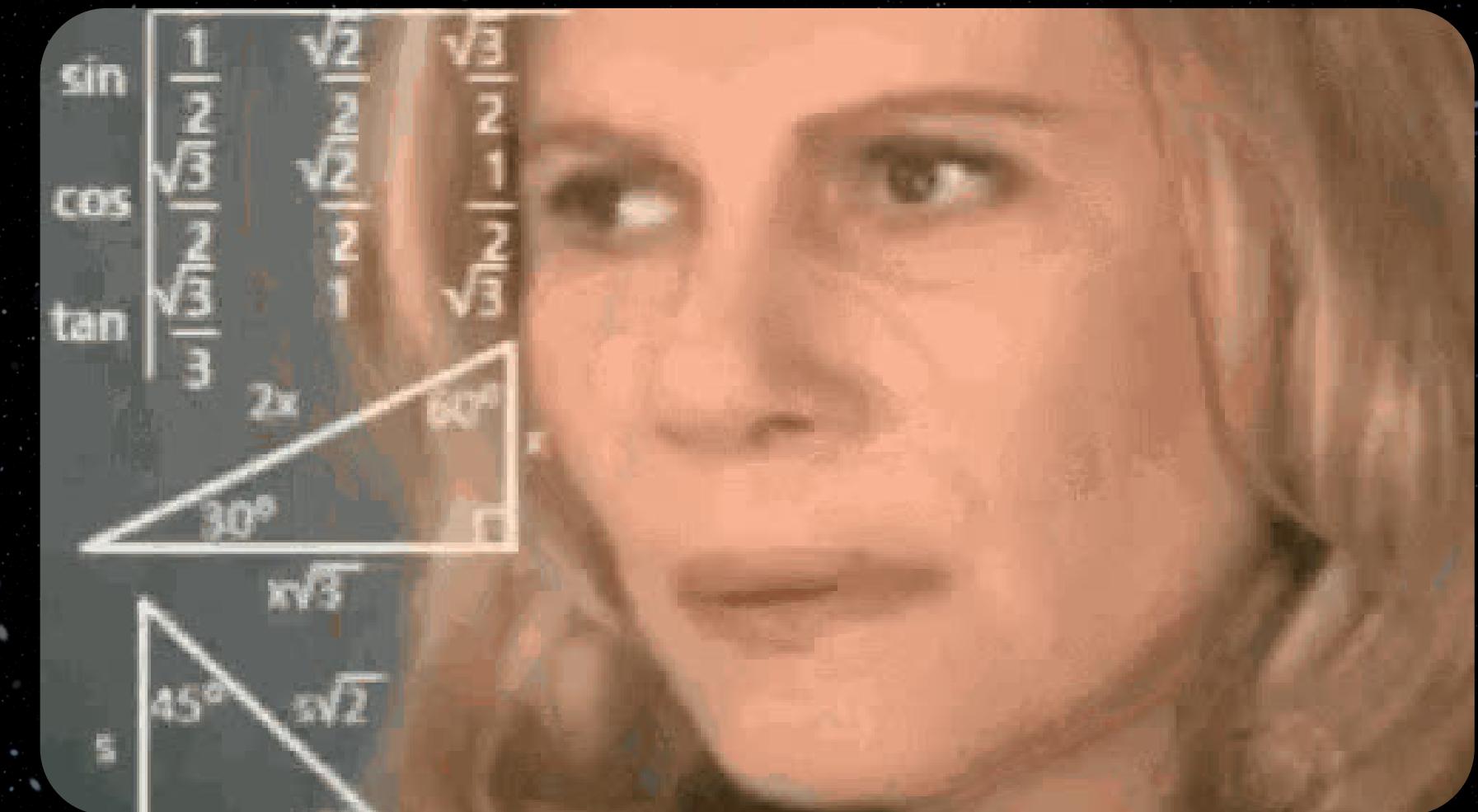
O Jogo da Imitação (2014)

Snowden: Herói ou Traidor (2016)

Enigma (2001)

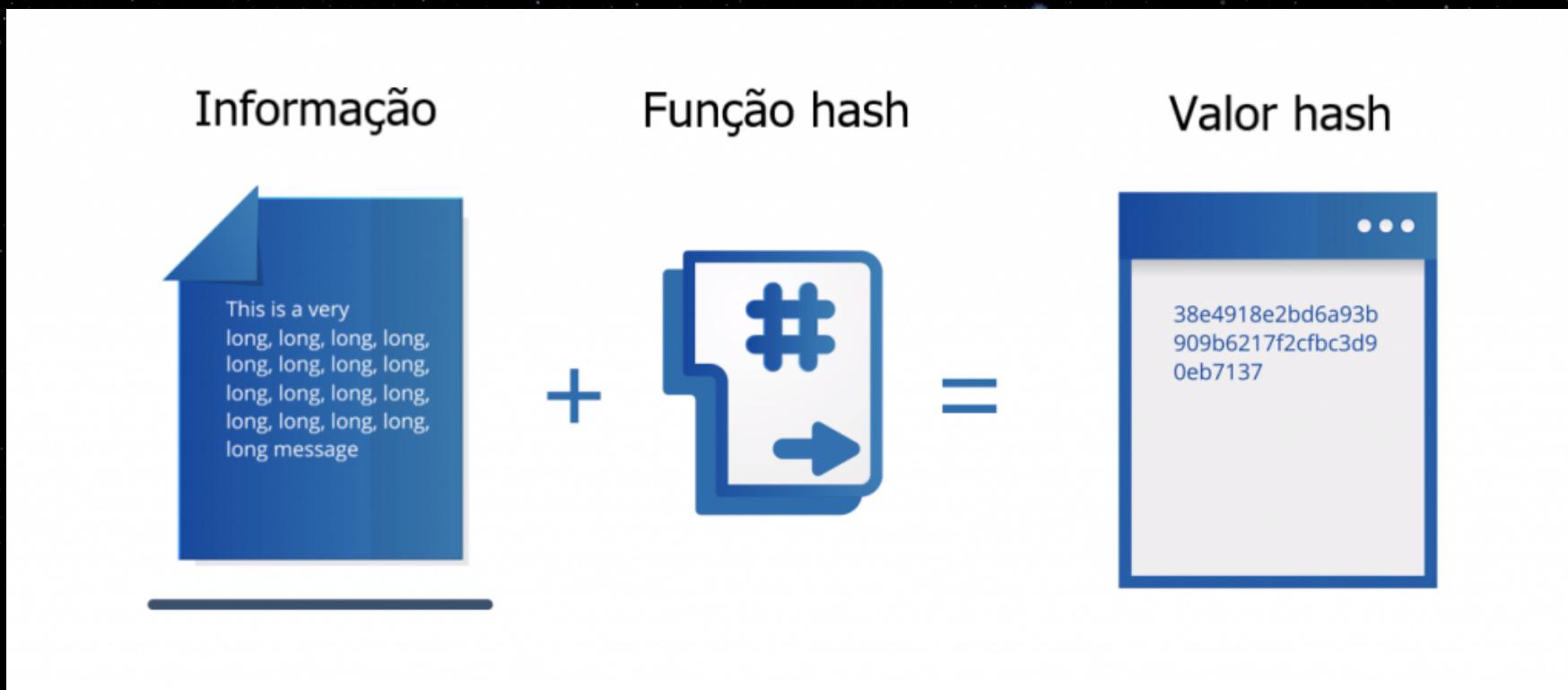
Uma Mente Brilhante (2001)

Lista de Filmes



HASH

Hash é uma string (texto) criptografada e é gerada a partir de uma função de Hash. Os algoritmos mais conhecidos para hash são: MD5, SHA-1 e SHA-2.



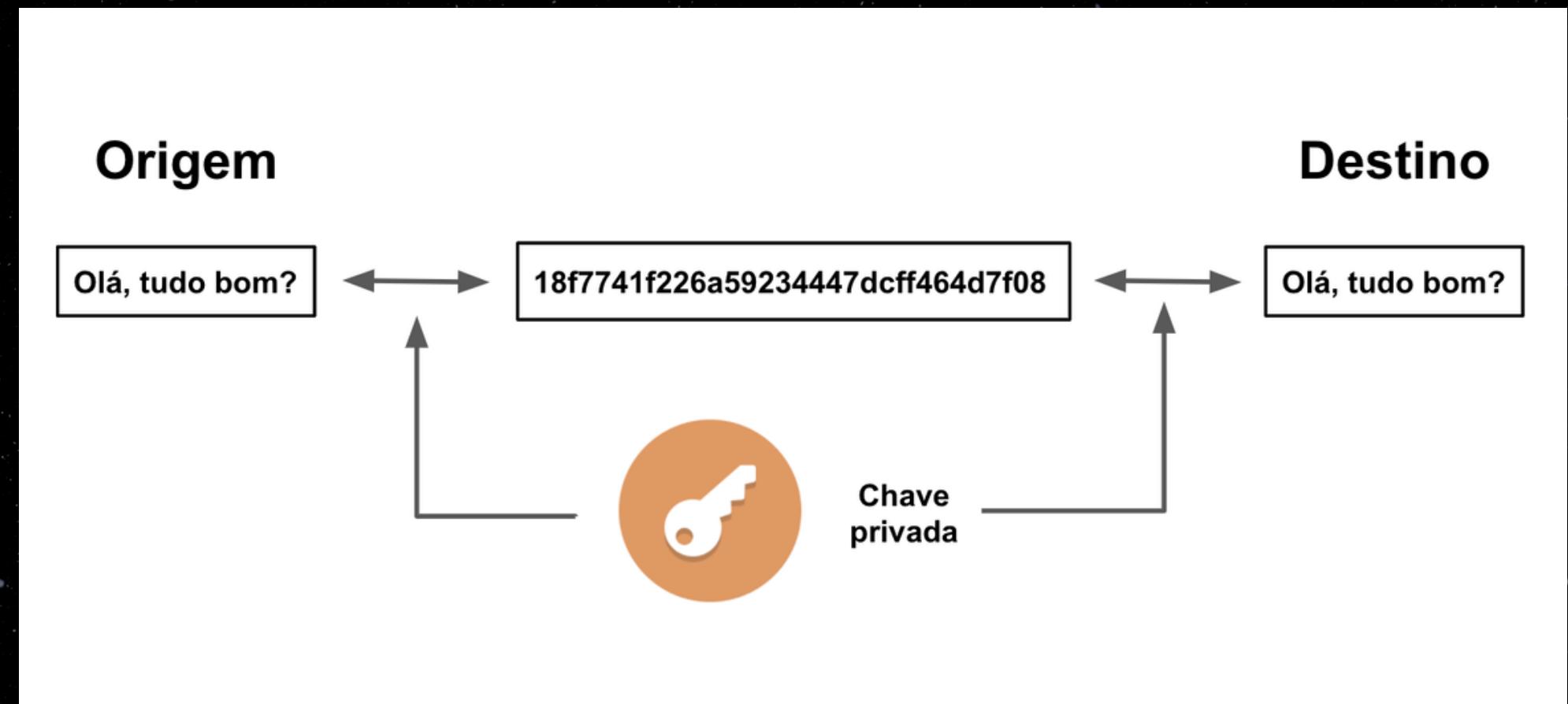
VANTAGEM:

Unidirecional, isto é, impossível de você voltar a string original a partir do hash.

DESVANTAGEM:

Não é possível recuperar uma senha; você só pode redefinir sua senha.

CRIPTOGRAFIA SIMÉTRICA



Os algoritmos de criptografia simétrica utilizam apenas uma chave para criptografar um dado qualquer, que pode ser uma mensagem, etc. Os algoritmos mais conhecidos são: DES, TripleDES, AES, RC4 e RC5.

CRIPTOGRAFIA SIMÉTRICA

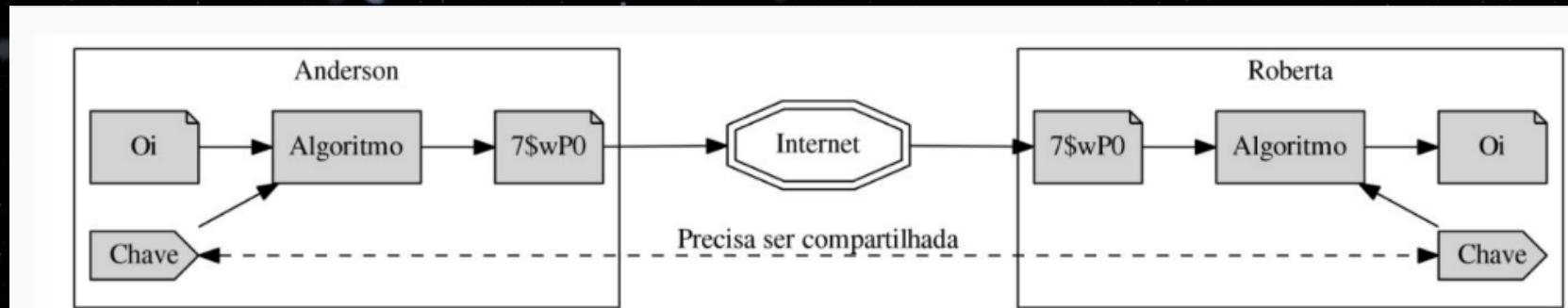


Figura 1. Envio de Dados com Criptografia Simétrica

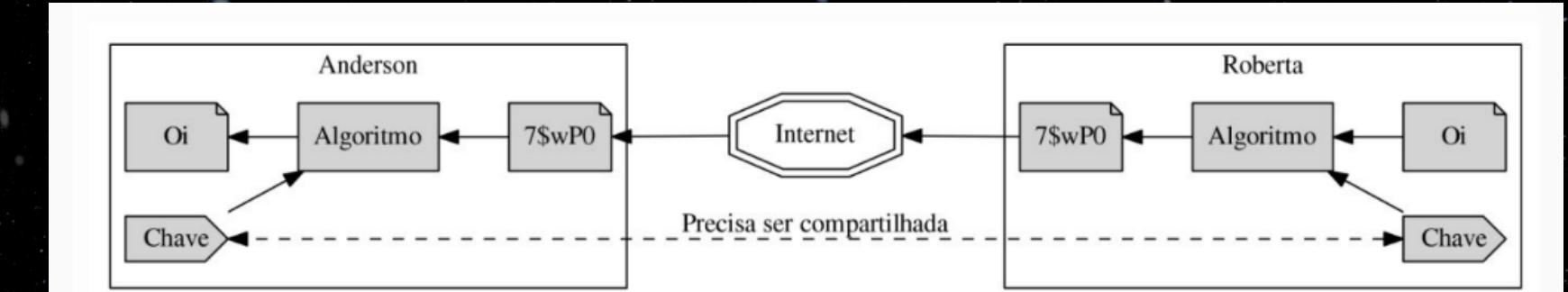


Figura 2. Recebimento de Dados com Criptografia Simétrica

VANTAGEM:

É muito mais rápido, o que traduz em baixa latência e pouco uso de CPU.

DESVANTAGEM:

Devido a utilização da mesma chave para criptografar e descriptografar, a chave precisa ser compartilhada com o receptor.

CRIPTOGRAFIA ASSIMÉTRICA



Criptografia assimétrica utilizam duas chaves complementares para criptografar e descriptografar. Uma das chaves é guardada em segredo e não é revelada a ninguém (chave privada) e outra pode ser publicada a qualquer um livremente (chave pública). Os algoritmos mais conhecidos são: RSA e ECDSA.

CRIPTOGRAFIA ASSIMÉTRICA

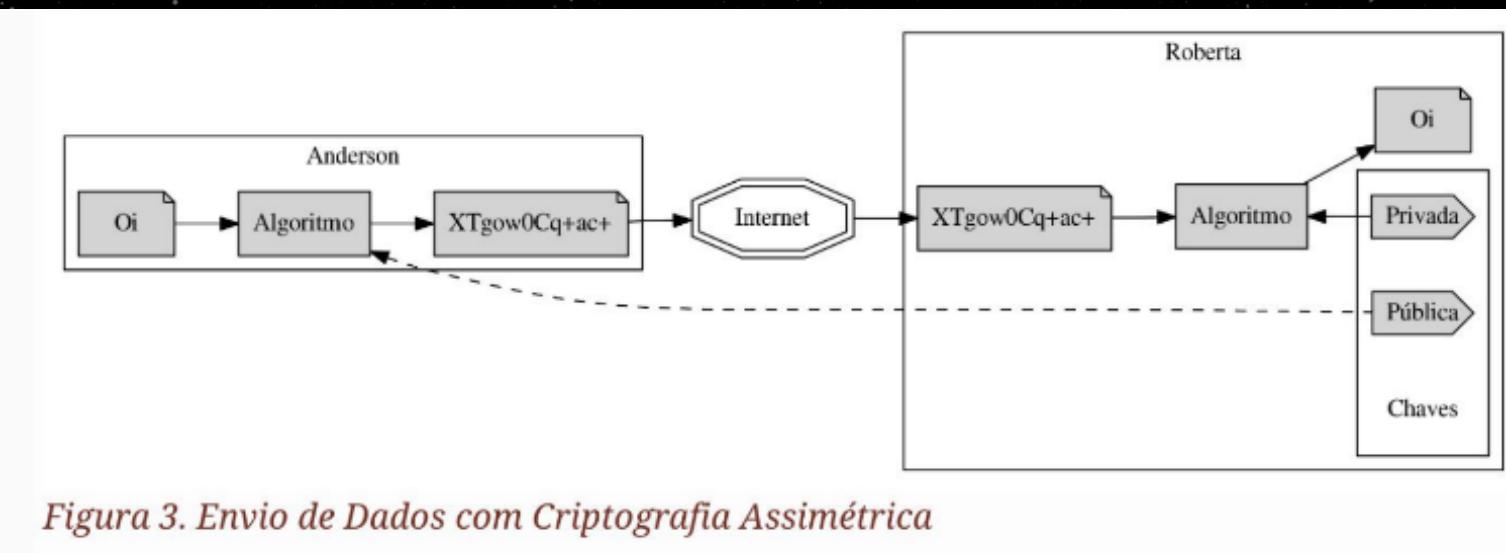


Figura 3. Envio de Dados com Criptografia Assimétrica

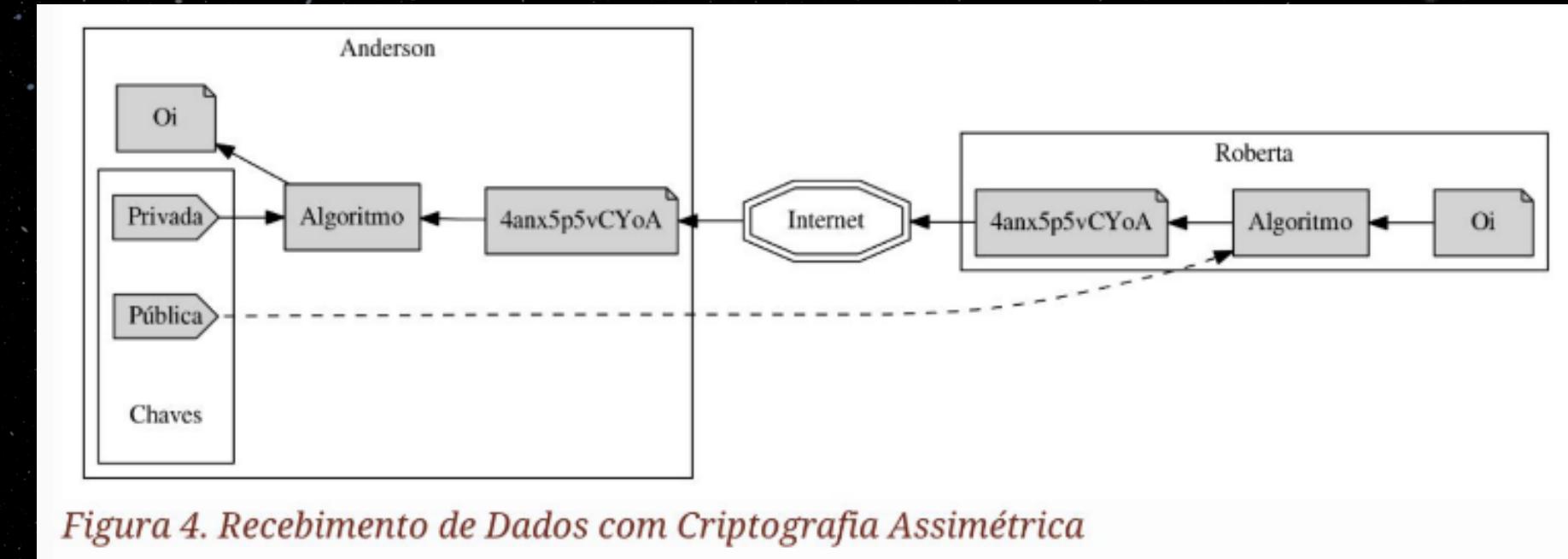


Figura 4. Recebimento de Dados com Criptografia Assimétrica

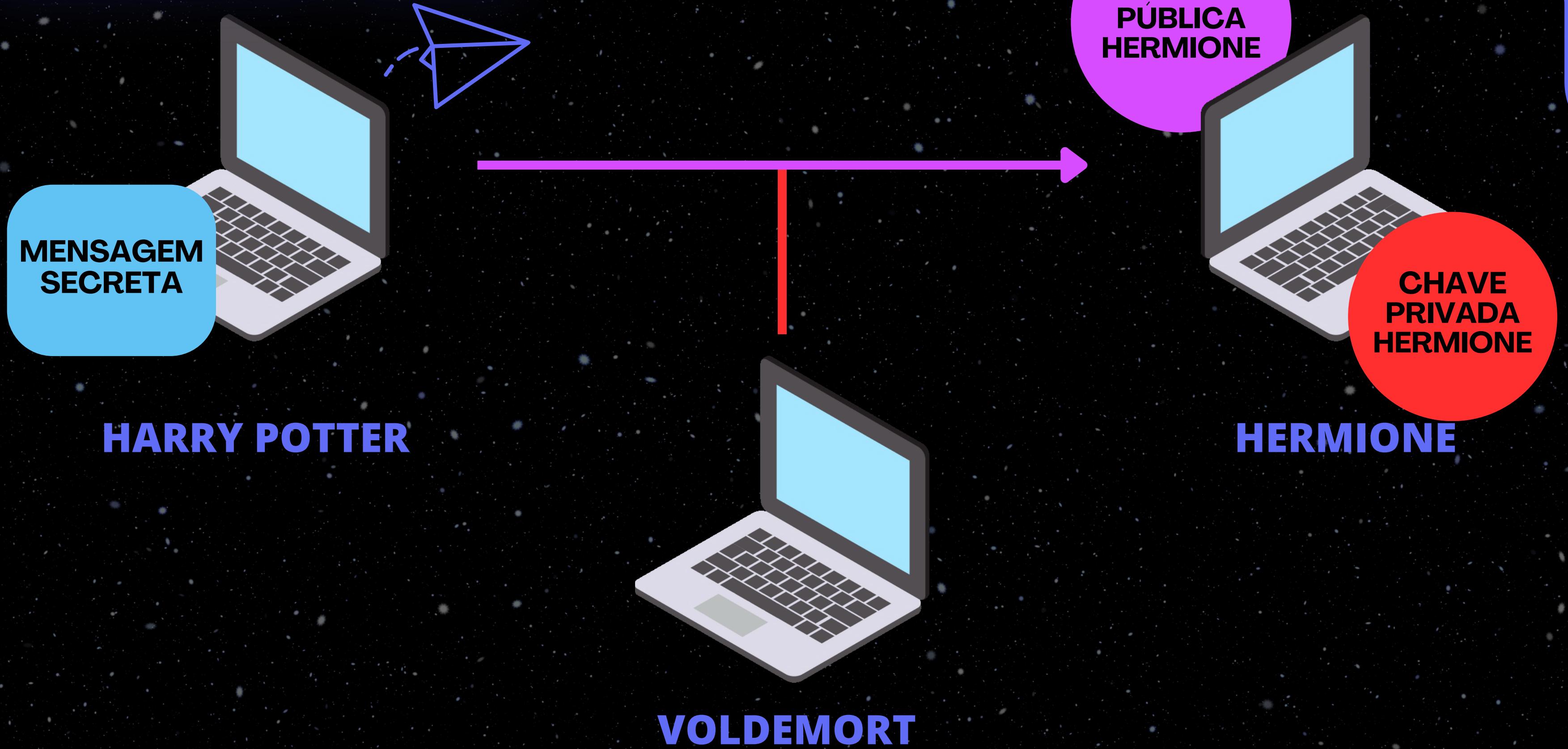
VANTAGEM:

É um dado criptografado com uma chave que pode apenas ser descriptografado com outra chave e vice-versa. Sendo assim uma comunicação segura, mesmo que o meio de comunicação não seja.

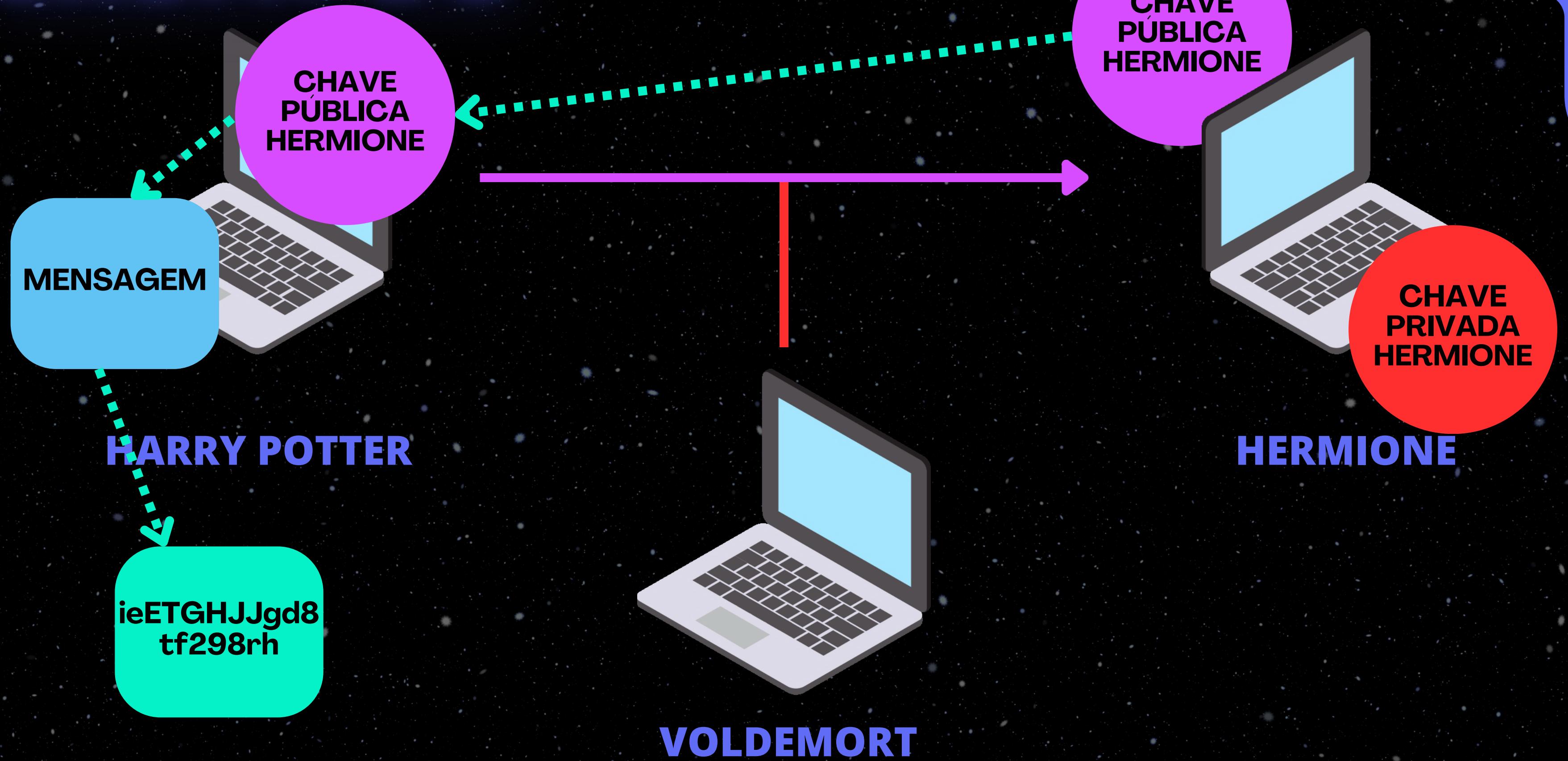
DESVANTAGEM:

São muito custosos em termos de CPU, por esse motivo as comunicações, normalmente, os utilizam como meio de troca de chave simétrica. Diminuindo, assim o tempo e recursos da CPU.

EXEMPLO



EXEMPLO



EXEMPLO

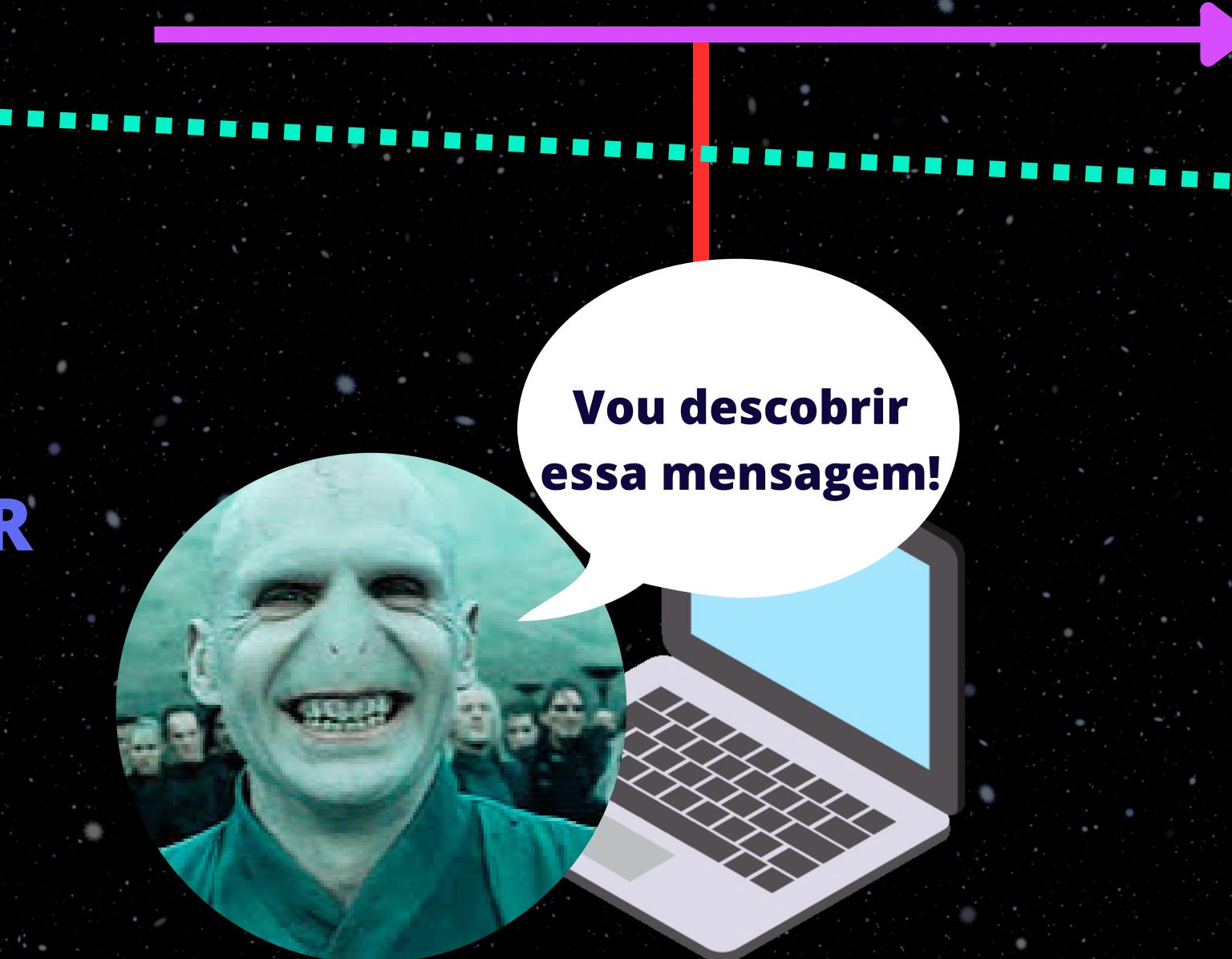


HARRY POTTER



VOLDEMORT

Vou descobrir
essa mensagem!



HERMIONE

EXEMPLO



HARRY POTTER



VOLDEMORT

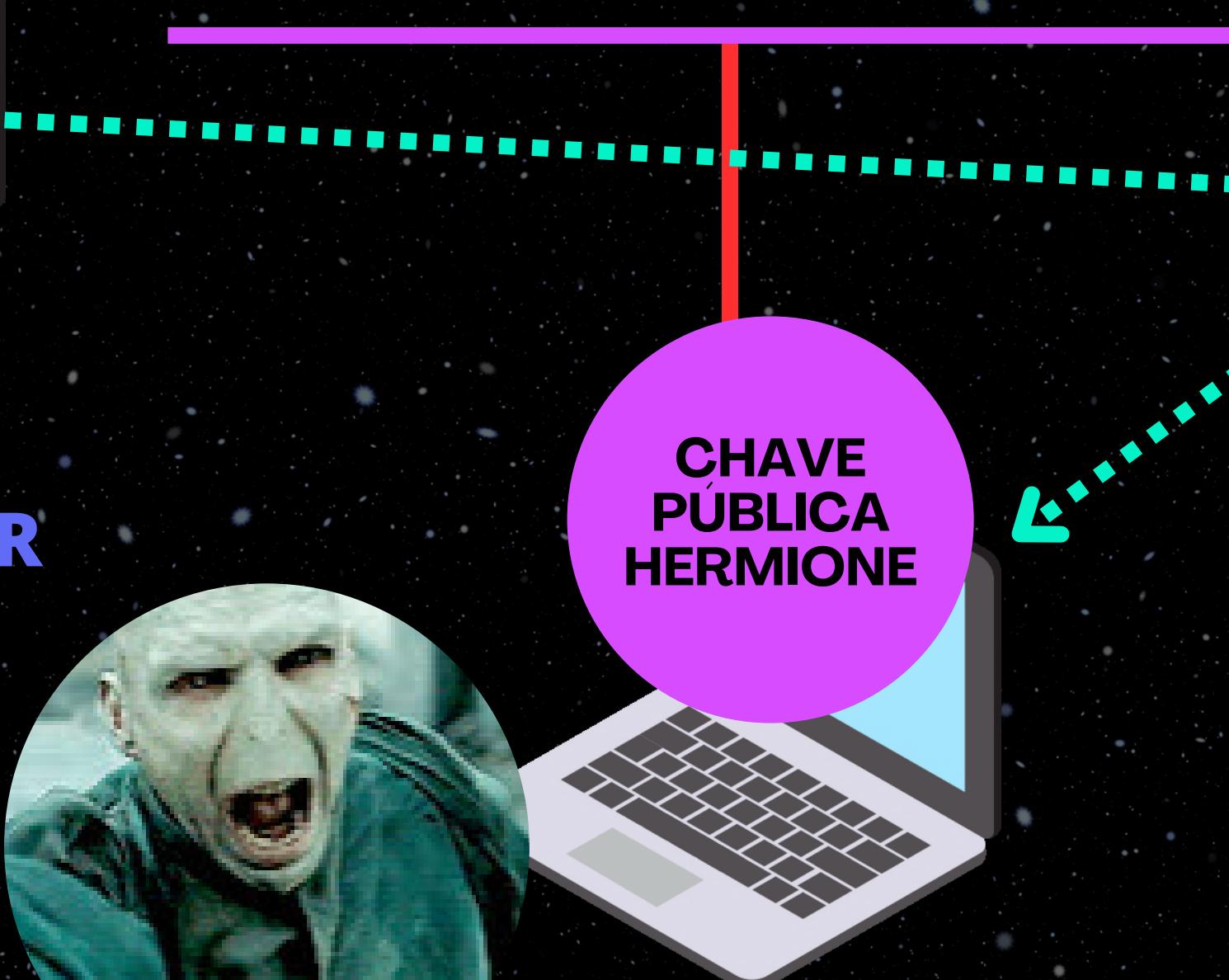
CHAVE
PÚBLICA
HERMIONE

CHAVE
PÚBLICA
HERMIONE

ieETGHJJgd8
tf298rh

CHAVE
PRIVADA
HERMIONE

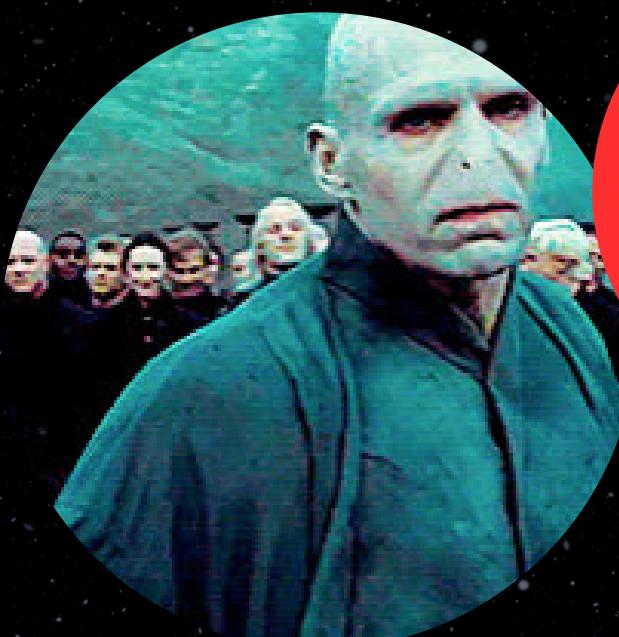
HERMIONE



EXEMPLO



HARRY POTTER



VOLDEMORT

CHAVE
PRIVADA
HERMIONE

CHAVE
PÚBLICA
HERMIONE

CHAVE
PÚBLICA
HERMIONE

ieETGHJJgd8
tf298rh

CHAVE
PRIVADA
HERMIONE

HERMIONE



Obs: A chave pública só é utilizada
nesse processo para criptografar.

ASSINATURAS

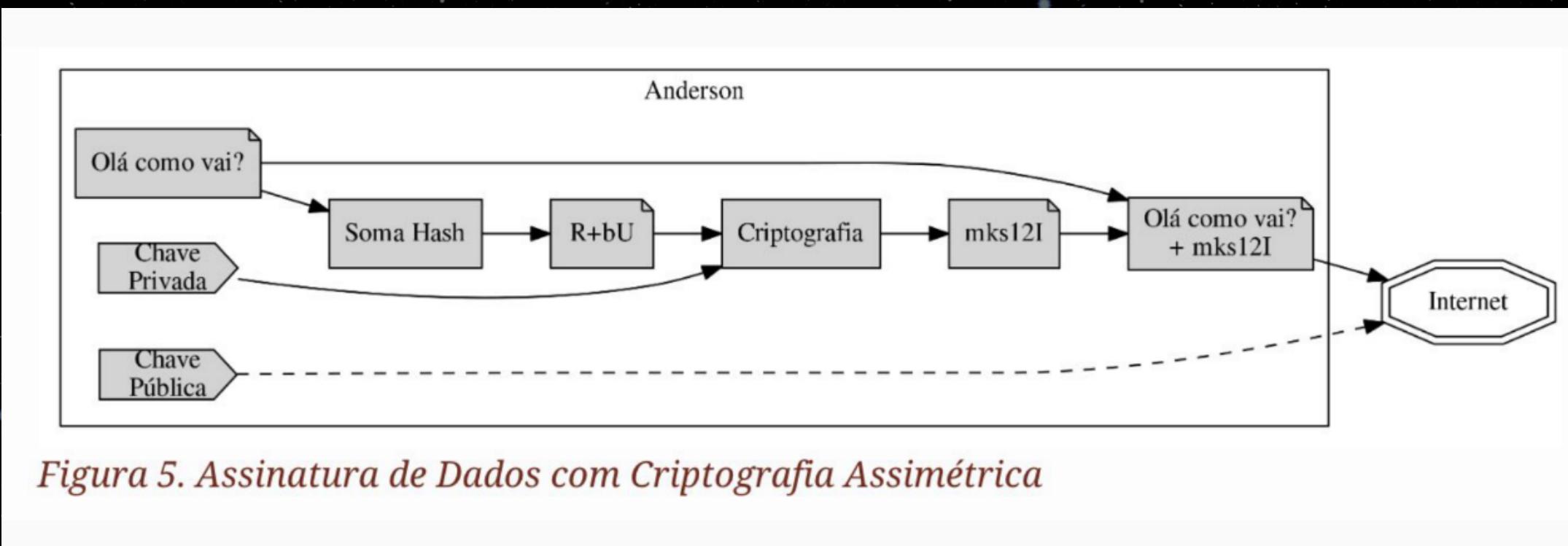
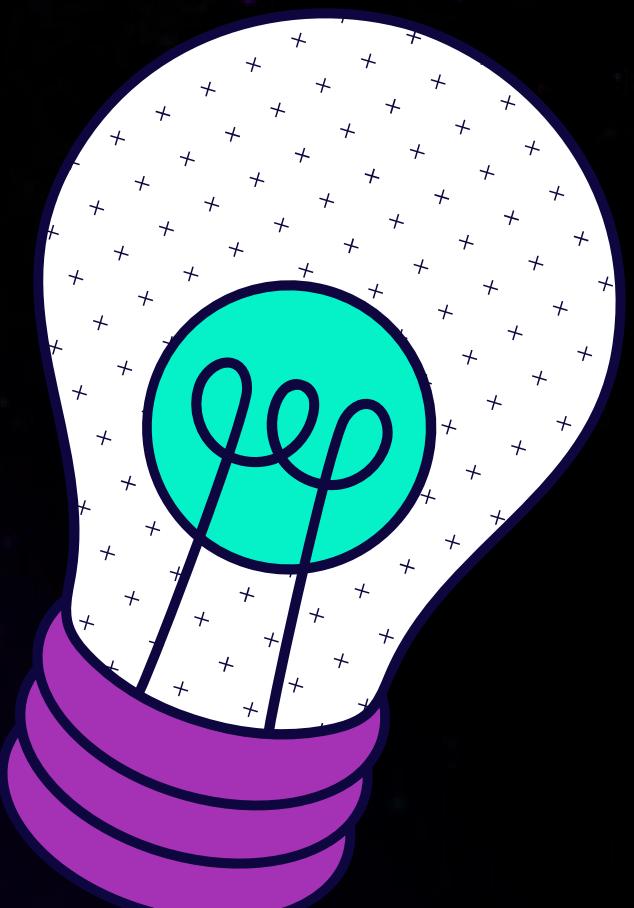
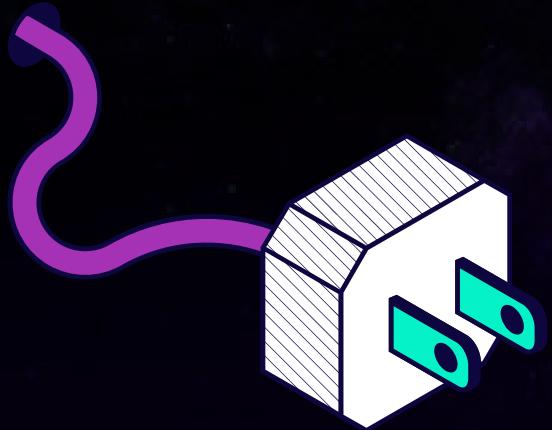


Figura 5. Assinatura de Dados com Criptografia Assimétrica

Um uso comum para a criptografia assimétrica (além de ser utilizada para garantir privacidade) são as assinaturas para garantir identidade.

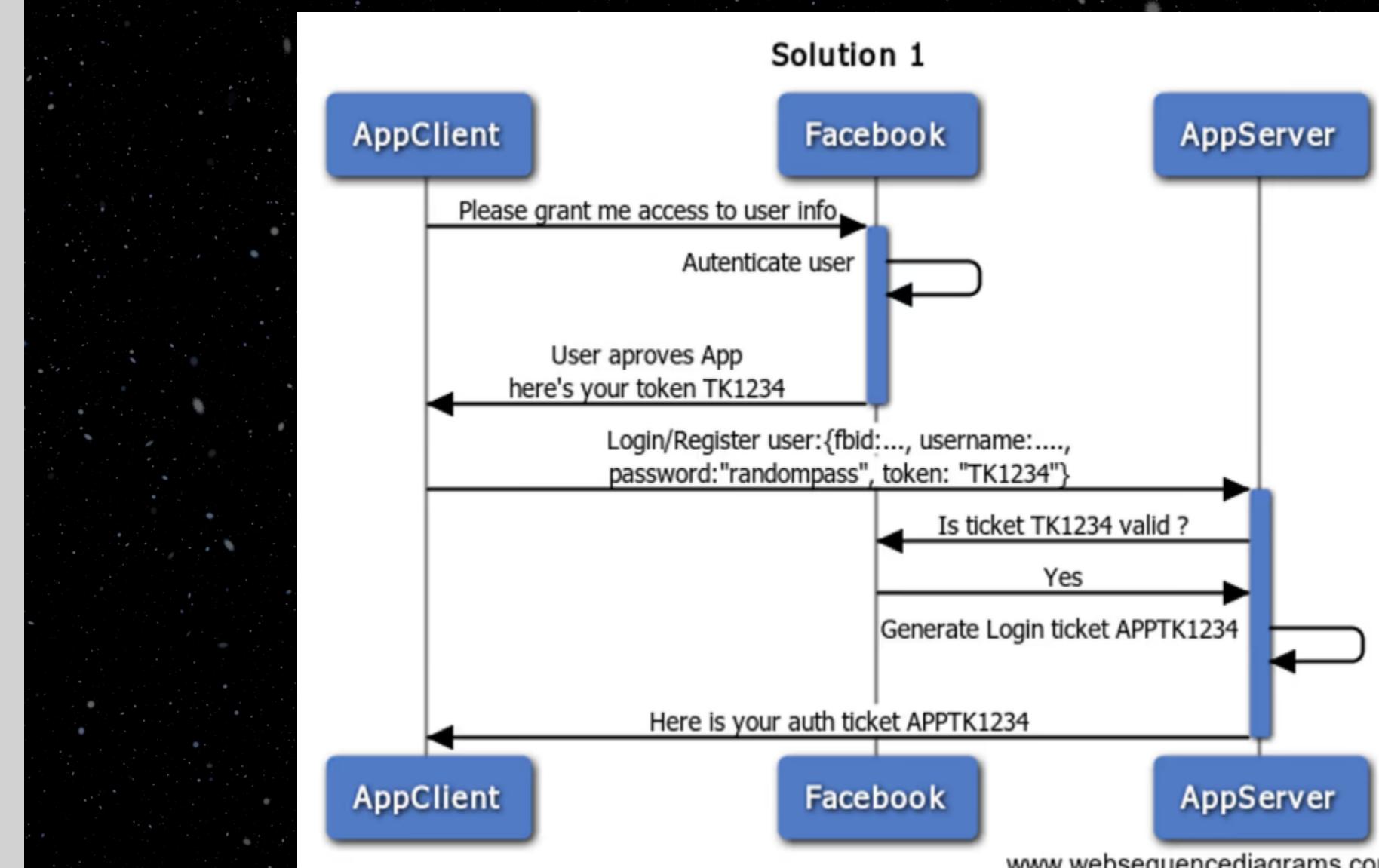
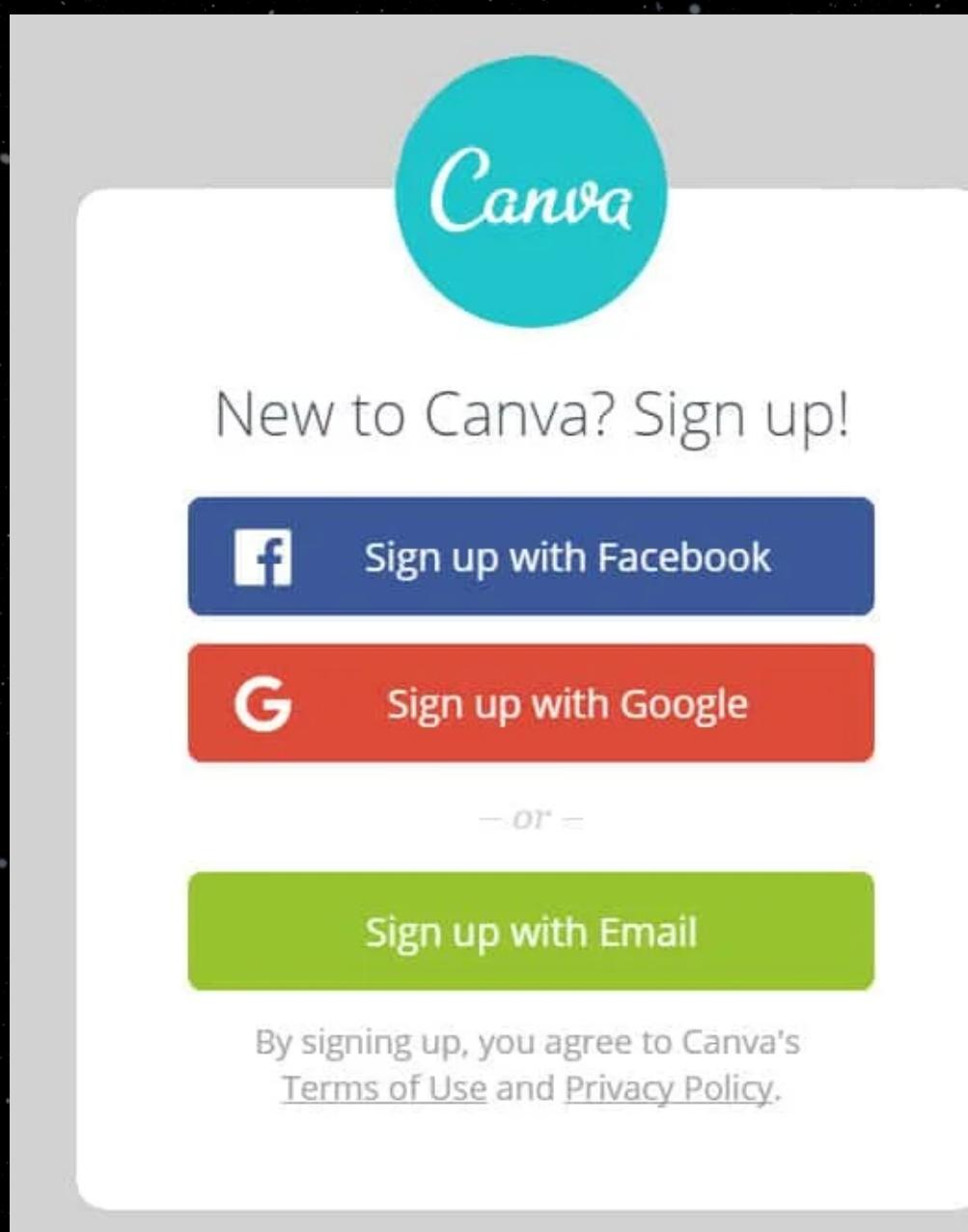
Uma maneira eficiente de alcançar o mesmo objetivo é gerar uma soma Hash (Checksum) do dado e criptografar esse resultado. Então a confirmação de identidade passaria a ser da seguinte maneira: gerar uma soma Hash do dado recebido, descriptografar a assinatura recebida e por fim comparar se os resultados são iguais.

Métodos de Autenticação



OAuth

É um mecanismo de autorização utilizado para realizar login por meio de redes sociais (ex: login pelo Facebook, Twitter etc).



JWT - JSON WEB TOKEN

Estrutura no formato Json, compacto e seguro, composto por chave/valor. Ele permite as informações sejam assinadas tanto com criptografia simétrica, quanto com criptografia assimétrica.

Envia para o servidor para que ele possa entender e disponibilizar os recursos que estamos querendo acessar.



ESTRUTURA DO JSON WEB TOKEN

DOCUMENTAÇÃO DO JWT

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9 . ey
JzdWIi0iIxMjM0NTY3ODkwIiwibmFtZSI6IkRvZ
3VpbmhvV2ViIiwiWF0IjoxNTE2MjM5MDIyfQ . e
HwTB3NbKWZMcg6bzYrgJb5Yrt-
fagDjsiKY4CBGs1k



HEADER



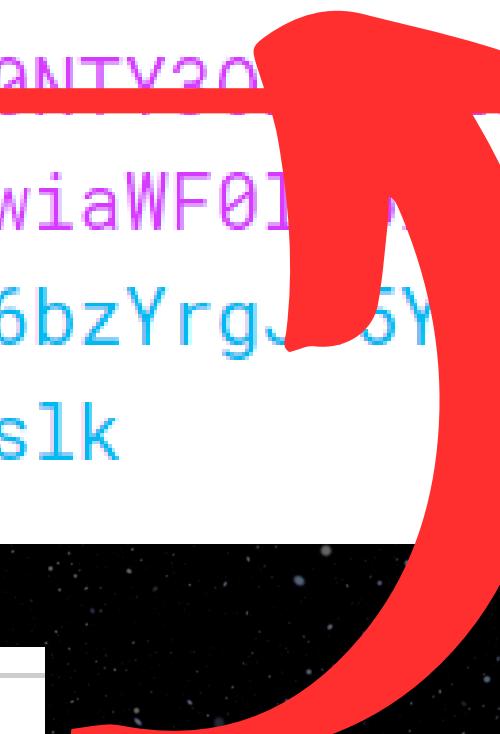
PAYOUT



VERIFY
SIGNATURE

Assim será a representação do seu token

ESTRUTURA DO JSON WEB TOKEN



```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9 ey  
IzdWTi0iTxMjM0NTY30  
3VpbmhvV2ViIiwiaWF0I  
HwTB3NbKWZMcg6bzYrgs5Y  
fagDjsiKY4CBGs1k
```

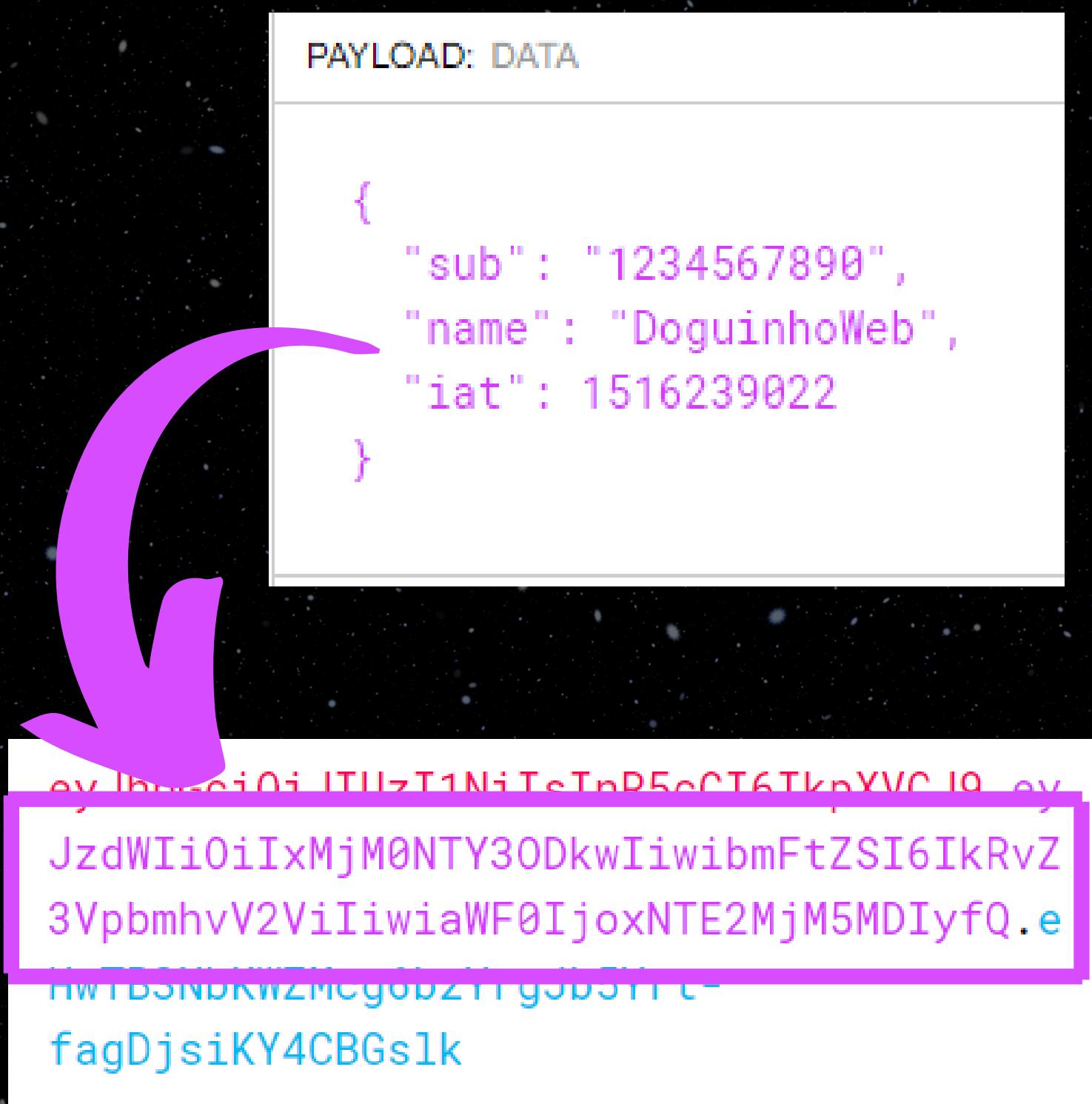
HEADER: ALGORITHM & TOKEN TYPE

```
{  
  "alg": "HS256",  
  "typ": "JWT"  
}
```

HEADER

Esse será o cabeçalho que define o tipo de algoritmo (a função que irá realizar para codificar/decodificar) e o tipo de token no caso o JWT.

ESTRUTURA DO JSON WEB TOKEN



PAYLOAD

São os dados que passamos no JWT entre as nossas requisições (geradas no Backend).

- "sub" é o id do usuário;
- "name" nome do usuário;
- "iat" marca a data de criação deste token, geralmente é acompanhado de uma chave chamada "exp" que diz quando irá expirar (encerrando a sessão do usuário);
- podemos passar também o "role" que define o que o usuário pode fazer no sistema. ex: "role": "admin".

ESTRUTURA DO JSON WEB TOKEN

VERIFY SIGNATURE

```
HMACSHA256(  
    base64UrlEncode(header) + "." +  
    base64UrlEncode(payload),  
    your-256-bit-secret  
)  secret base64 encoded
```

VERIFY SIGNATURE

É a assinatura, onde temos a validação do token para a aplicação.

SECRET - segredo para ajudar na criptografia (é o que diferencia cada aplicação), geralmente uma string bem forte como se fosse uma senha (é onde está a maior parte da segurança, por ser única e diferente em cada aplicação)

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9 . ey
WIi0iIxMjM0NTY3ODkwIiwibmFtZSI6IkRvZ
mbhvV2ViTiwiawEATjoxNTE2MiM5MDTyfQ . e

HwTB3NbKWZMcg6bzYrgJb5Yrt-
fagDjsiKY4CBGs1k

Vamos exercitar o que aprendemos?

Mão na massa!

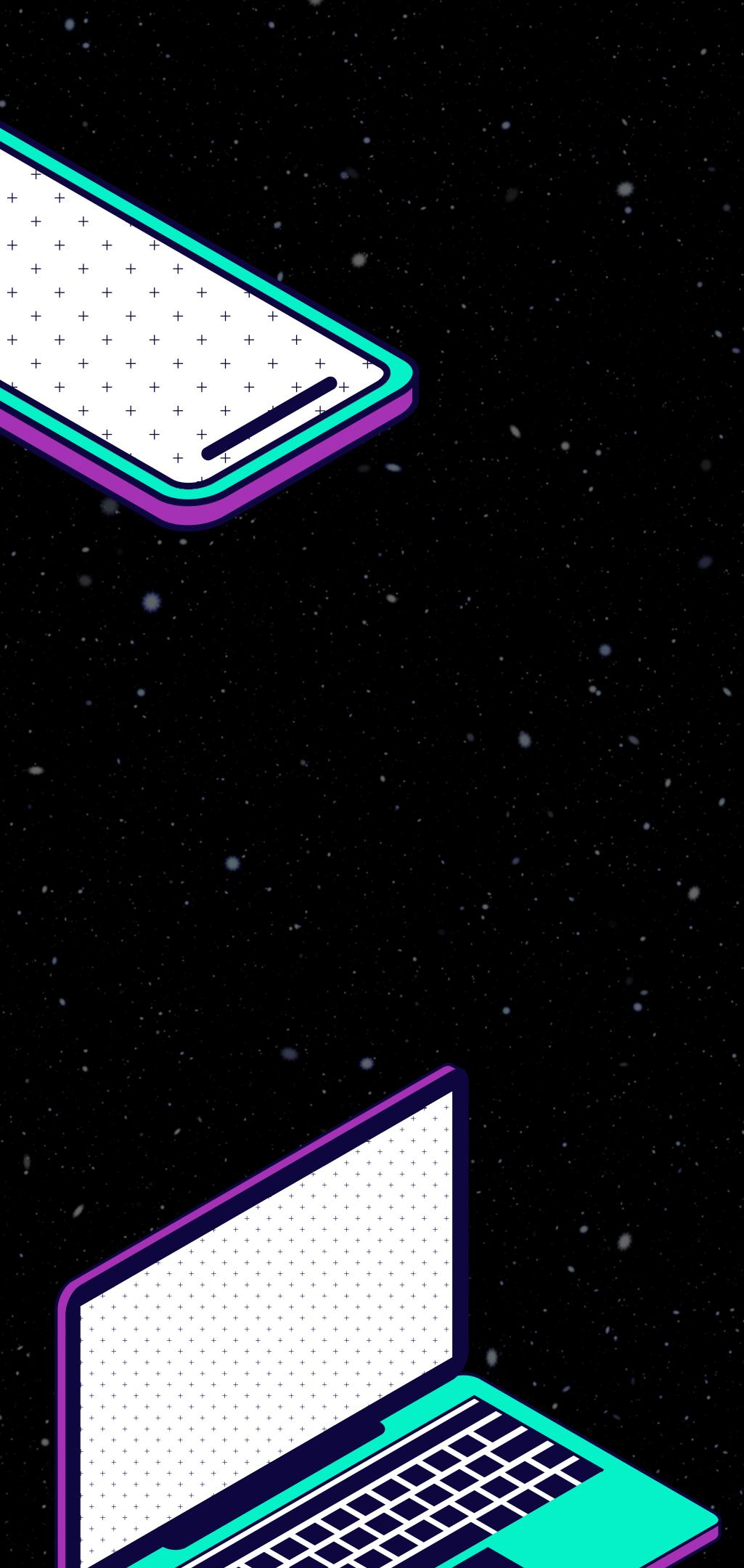




Intervalo! - Retorno às -



{ Reprograma }



Obrigada!
Tenha um ótimo
final de semana.

E fiquem orgulhosas por cada passo desse
processo!



{ Reprogramma }