



# Recolha Autônoma

Web API – Manual Técnico



## HISTÓRICO DE MODIFICAÇÕES

Revisão	Data Modificação	Descrição da modificação	Modificado por
0	31/10/2019	Versão inicial	Tecnologia da Informação – Squad Rodoviário
1	17/01/2019	Adicionando tabela de erros  Simplificando a descrição do processo de autenticação  Adicionando URLs de produção e homologação	Tecnologia da Informação – Squad Rodoviário
2	24/03/2020	Inclusão atributo numeroAutorizacao no serviço “Consultar código de autorização”  Inclusão do erro ERA000008 na tabela de erros	Tecnologia da Informação – Squad Rodoviário
3	26/03/2020	Ajuste do path do serviço de consulta de informações para emissão de NFE  Removido exemplo default do serviço de consulta de informações para emissão de NFE  Revisão geral na seção Erros, renomeada para Validações da API Ajustada definição do parâmetro “numeroTransacao” no serviço de consulta de código de autorização  Incluído detalhamento do “codigolbge” no serviço de consulta de informações para emissão de NFE  Substituição de nomes dos revisores pela área responsável pela revisão na seção de histórico de modificações	Tecnologia da Informação – Squad Rodoviário

## HISTÓRICO DE MODIFICAÇÕES

Revisão	Data Modificação	Descrição da modificação	Modificado por
4	31/03/2020	Ajustada descrição do item "volumeAbastecido" no serviço de consulta de código de autorização  Adicionado exemplo de data no formato ISO8601 no serviço de consulta de código de autorização	Tecnologia da Informação – Squad Rodoviário
5	04/06/2020	Adicionado o endpoint Upload Nota Fiscal  Inclusão de erros no (Erros API) do ERA000009 até ERA000024	Tecnologia da Informação – Squad Rodoviário
6	16/06/2020	Alteração WS Consultar informações para emissão de NFE:  - Inclusão do parâmetro "dataAutorizacao"  - Regra no parâmetro "codigoAutorizacao" é o código da transação ou o número de autorização  - Retirada do erro ERA000009	Tecnologia da Informação – Squad Rodoviário

The background image shows a close-up of hands typing on a laptop keyboard. The scene is overlaid with a futuristic digital aesthetic, featuring glowing teal and yellow lines, hexagonal patterns, and circular motifs. A prominent teal circle is in the top right, and another is in the bottom center. A white rounded rectangle is in the top right, and a white circle is in the top left. The text is positioned in a teal rounded rectangle on the left side.

**Recolha Autônoma**  
Web API – Manual Técnico

# Sumário

Introdução .....	6
Autenticação .....	7
Paths .....	8
Consultar código de autorização .....	8
Consultar informações para emissão de NFE ....	12
Upload de Nota Fiscal .....	16
Validações da API .....	19
Orientações e insumos para os testes .....	20

# Introdução

As APIS da Recolha Autônoma, são do tipo API REST, com objetivo de fornecer as seguintes possibilidades para o usuário dos serviços:

## **Consultar código de autorização URL Produção:**

<https://srv1.ticketlog.com.br/ticketlog-servicos/credenciamento>

URL Homologação:

<https://serviceshtml.ticketlog.com.br/ticketlog-servicos/credenciamento>

## **Consultar informações para emissão de NFE URL Produção:**

<https://srv1.ticketlog.com.br/ticketlog-servicos/credenciamento>

URL Homologação:

<https://serviceshtml.ticketlog.com.br/ticketlog-servicos/credenciamento>

## **Upload de Nota Fiscal URL Produção:**

<https://srv1.ticketlog.com.br/ticketlog-servicos/credenciamento>

URL Homologação:

<https://serviceshtml.ticketlog.com.br/ticketlog-servicos/credenciamento>



# Autenticação

## basicAuth

A Autenticação para utilização da API deve ser executada através do esquema **“HTTP Basic”**. Neste esquema de autenticação, é utilizado um par de login e senha.

Geralmente as bibliotecas cuidam do envio do usuário e senha, ou seja, basta informar um nome de usuário e uma senha válidos e a biblioteca fará todo o resto.

Caso este não seja o caso, poderá implementar a autenticação manualmente. Para isto basta enviar, em cada requisição, um cabeçalho HTTP com o nome **“Authorization”**.

Um exemplo para o valor deste cabeçalho é: Basic dGVzdGU6dGVzdGU=  
Onde **“Basic”** é uma palavra que sempre será enviada.

O restante **“dGVzdGU6dGVzdGU=”**, é o resultado da codificação em base64, da string **“teste:teste”**. Neste caso, o primeiro **“teste”** é o nome de usuário. O segundo **“teste”** é a senha. No esquema de Autenticação **“Basic”**, o símbolo “:” (dois pontos) tem a função de separar o usuário da senha.

Portanto, constatamos que, para um par de login/senha onde o Login seja **“teste”** e a Senha também seja **“teste”**, temos o cabeçalho final: Basic dGVzdGU6dGVzdGU=

Para mais informações sobre este esquema de autenticação, visitar:  
[https://en.wikipedia.org/wiki/Basic\\_access\\_authentication](https://en.wikipedia.org/wiki/Basic_access_authentication)

# Paths

## Consultar código de autorização

### PATH

**POST** /recolhaAutonoma/buscarAutorizacao

### DESCRIÇÃO

Consultar o código de autorização conforme os parâmetros informados para a consulta.

**REQUEST** [Content-Type: application/json]

**codigoEstabelecimento:** integer (int64) **requerido**

Código Ticketlog do estabelecimento onde ocorreu a transação de abastecimento.

**valorCupom:** number (double - 2 casas decimais) - **requerido**

Valor em R\$ no formato indicado para transação de abastecimento.

**dataHoraCupom:** date (yyyy-MM-dd'T'HH:mm:ss'Z') - **requerido**

Data e hora no formato ISO 8601 para transação de abastecimento.  
Exemplo: 2020-03-01T10:00:00-03:00

**volumeAbastecido:** number (double - 2 casas decimais) - **requerido**

Volume em litros realizado na transação de abastecimento.

**numeroAutorizacao:** integer (int64)



Em casos em que o integrador possui o número de autorização emitido pelo meio de captura, seja ele POS ou TEF, e deseje validar este número, basta submeter o número de autorização neste parâmetro.

Quando enviado, este número será validado frente aos demais parâmetros do serviço a fim de identificar a transação desejada, ou seja, caso o sistema encontre mais de 1 registro para os parâmetros informados o númeroAutorizacao será o critério de desempate.

Se o número enviado estiver coerente com os demais parâmetros, a API vai validar as informações e retornar o mesmo número no retorno da chamada como **"codigoAutorizacao"**. Indicando assim que o número informado pelo integrador está correto para a transação indicada.

Caso não esteja coerente, a API vai validar as informações e retornar no **"codigoAutorizacao"** o código de autorização correto para a transação desejada (frente aos demais parâmetros de valor, horário e volume). Indicando assim que o número informado pelo integrador não está correto para a transação indicada.

```
"codigoEstabelecimento": "integer",  
"valorCupom": "number",  
"dataHoraCupom": "date",  
"volumeAbastecido": "number",  
"numeroAutorizacao": "integer"
```

Exemplo default

### RESPONSE

[Content-Type: application/json]

**codigoAutorizacao:** integer (int64)

Código da autorização que atendeu aos parâmetros informados na requisição.

**erros:** array

Lista de erros encontrados nas validações de dados ou de processamento da API.

**Codigo:** string (9 caracteres máx.)

Código único do erro identificado pela API

**Propriedade:** string (255 caracteres máx.)

Nome da propriedade do JSON de entrada a qual o erro está associado.

**detalhe:** string (255 caracteres máx.)

Informação detalhada sobre o erro encontrado.

### HTTP STATUS CODES

**200** – OK (Autorização encontrada)

**400** – Bad Request (Erros nos dados informados)

**404** – Not Found (Não foi encontrada autorização para os dados informados)

**460** – Business Exception (Erros referentes as regras operacionais do sistema de recolha)

`"codigoAutorizacao": "integer"`

Exemplo http status code: 200

```
{
  "erros": [
    {
      "codigo": "string",
      "propriedade": "string",
      "detalhe": "string"
    }
  ]
}
```

Exemplo http status code: 400



# Consultar informações para emissão de NFE

## PATH

GET

/recolhaAutonoma/dadosEmissaoNfeRecolhaAutonoma

## DESCRIÇÃO

Consultar as informações para emissão de NFE com base no código de autorização.

## REQUEST [Query Params]

**codigoEstabelecimento:** integer (int64) **requerido**

Código Ticketlog do estabelecimento onde ocorreu a transação de abastecimento.

**codigoAutorizacao:** integer (int64) **requerido**

Código da autorização para qual deseja-se consultar os dados de emissão da NFE.

O código de autorização único da TicketLog para o abastecimento em questão, pode ser obtido utilizando o serviço "Consultar código de autorização" descrito neste manual. Adicionalmente, o serviço permite informar neste campo o número de autorização emitido pelos meios de captura (POS ou TEF) no momento do abastecimento, caso esta opção seja escolhida para integração, então o campo **dataAutorizacao** será obrigatório.

**dataAutorizacao:** date (yyyy-MM-dd')

Data no formato ISO 860-1 para a transação de abastecimento. A data é obrigatória conforme regras do campo **codigoAutorizacao**, e quando informada, não pode ser maior que a atual.

Exemplo: 2020-03-01

### RESPONSE

[Content-Type: application/json]

**cnpj:** string (14 caracteres máx.)

CNPJ do cliente (somente números) para emissão da NFE.

**razaoSocial:** string (80 caracteres máx.)

Razão social do cliente para emissão da NFE.

**inscricaoEstadual:** string (20 caracteres máx.)

Inscrição estadual do cliente para emissão da NFE.

**cfop:** string (35 caracteres máx.)

Lista de CFOPs definidos pelo cliente para emissão da NFE.

**endereco:** element

Agrupador das informações do endereço.

**logradouro:** string (80 caracteres máx.)

Lagradoiro do endereço de emissão da NFE.

**numero:** string (6 caracteres máx.)

Número do endereço de emissão da NFE, quando sem número exhibe "S/N".

**complemento:** string (20 caracteres máx.)

Complemento do endereço de emissão da NFE.

**bairro:** string (60 caracteres máx.)

Nome do bairro do endereço de emissão da NFE.

**cidade:** string (50 caracteres máx.)

Nome da cidade do endereço de emissão da NFE.

**uf:** string (2 caracteres)

Sigla da UF do endereço de emissão da NFE.

**cep:** string (8 caracteres)

cep (somente números) do endereço de emissão da NFE.

**codigolbge:** integer (int64)

Código da cidade conforme base do IBGE.

**erros:** array

Lista de erros encontrados nas validações de dados ou de processamento da API.

**codigo:** string (9 caracteres máx.)

Código único do erro identificado pela API

**propriedade:** string (255 caracteres máx.)

Nome da propriedade do JSON de entrada a qual o erro está associado.

**detalhe:** string (255 caracteres máx.)

Informação detalhada sobre o erro encontrado.

### HTTP STATUS CODES

**200** – OK (Autorização encontrada)

**400** – Bad Request (Erros nos dados informados)

**404** – Not Found (Não foi encontrada autorização para os dados informados)

**460** – Business Exception (Erros referentes as regras operacionais do sistema de recolha)

```
"cnpj": "string",
"razaoSocial": "string",
"inscricaoEstadual": "string",
"cfop": "string",
"endereco" {
  "logradouro": "string",
  "numero": "string",
  "complemento": "string",
  "bairro": "string",
  "cidade": "string",
  "uf": "string",
  "cep": "string",
  "codigolbge": "integer"
```

Exemplo http status code: 200

```
{
  "erros": [
    {
      "codigo": "string",
      "propriedade": "string",
      "detalhe": "string"
    }
  ]
}
```

Exemplo http status code: 400



# Upload de Nota Fiscal

## PATH

**POST** /recolhaAutonoma/nfe

## DESCRIÇÃO

Possibilitar que o estabelecimento realize o envio do arquivo XML de nota fiscal emitida para um cliente que utiliza o sistema de recolha de notas.

**REQUEST** [Content-Type: application/json]

**codigoEstabelecimento:** integer (int64) **requerido**

Código Ticketlog do estabelecimento onde ocorreu a transação de abastecimento.

**codigoAutorizacao:** integer (int64) **requerido**

Código da autorização para qual deseja-se consultar os dados de emissão da NFE.

O código de autorização único da TicketLog para o abastecimento em questão, pode ser obtido utilizando o serviço “Consultar código de autorização” descrito neste manual. Adicionalmente, o serviço permite informar neste campo o número de autorização emitido pelos meios de captura (POS ou TEF) no momento do abastecimento, caso esta opção seja escolhida para integração, então o campo **dataAutorizacao** será obrigatório.

**dataAutorizacao:** date (yyyy-MM-dd')

Data no formato ISO 860-1 para a transação de abastecimento. A data é obrigatória conforme regras do campo **codigoAutorizacao**, e quando informada, não pode ser maior que a atual.

Exemplo: 2020-03-01

**arquivo.nomeOriginal:** string (60 caracteres máx.) - **requerido**

Nome original do arquivo que está sendo enviado na chamada, incluindo a extensão. (.xml).

**arquivo.base64Encode:** string (1000000 caracteres máx.) - **requerido**

Conteúdo do arquivo XML da nota fiscal com codificação base64/ISO 8859-1. O tamanho máximo do arquivo decodificado deve ser de 1000 Kb. Será validado utilizando o hash MD5, e se o conteúdo representa um arquivo XML.

**arquivo.MD5hash:** string (32 caracteres máx.) - **requerido**

Valor codificado hash MD5 do arquivo XML enviado no campo **base64Encode**.

```
{
  "codigoEstabelecimento": "integer",
  "codigoAutorizacao": "number",
  "dataAutorizacao": "date",
  "arquivo": {
    "nomeOriginal": "string",
    "base64Encode": "string",
    "MD5hash": "string"
  }
}
```

Exemplo default

### RESPONSE

[Content-Type: application/json]

**erros:** array

Lista de erros encontrados nas validações de dados ou de processamento da API.

**codigo:** string (9 caracteres máx.)

Código único do erro identificado pela API

**propriedade:** string (255 caracteres máx.)

Nome da propriedade do JSON de entrada a qual o erro está associado.

**detalhe:** string (255 caracteres máx.)

Informação detalhada sobre o erro encontrado.

### HTTP STATUS CODES

**200** – OK (Autorização encontrada)

**400** – Bad Request (Erros nos dados informados)

**404** – Not Found (Não foi encontrada autorização para os dados informados)

**460** – Business Exception (Erros referentes as regras operacionais do sistema de recolha)

```
{
  "erros": [
    {
      "codigo": "string",
      "propriedade": "string",
      "detalhe": "string"
    }
  ]
}
```

Exemplo http status code: 400

## Validações da API

Abaixo segue a tabela de validações executada pela API, quando pertinente, a API retorna as exceções abaixo vinculando código e descrição no corpo da resposta (JSON) com http status code: 460

Código	Descrição
ERA000001	A propriedade deve ser preenchida
ERA000002	O valor deve ser maior que zero
ERA000003	A data/hora deve ser igual ou anterior à data/hora atual
ERA000004	O valor deve ter no máximo dois dígitos de precisão
ERA000005	Estabelecimento não encontrado ou desativado
ERA000006	Cliente não encontrado ou desativado
ERA000007	Cliente não possui recolha de notas
ERA000008	A transação indicada nos parâmetros da chamada foi encontrada, porém o cliente não possui serviço de recolha de notas contratado
ERA000010	Transação não encontrada para número de autorização e data informados
ERA000011	Erro inesperado ao buscar a transação via número de autorização
ERA000012	O ano da data deve estar entre 1753 e 9999
ERA000013	O conteúdo não representa uma codificação base64 válida
ERA000014	Erro inesperado ao verificar o tipo do arquivo enviado
ERA000015	O tipo de arquivo enviado não é XML
ERA000016	Erro inesperado ao verificar a hash do arquivo enviado
ERA000017	A hash MD5 enviada não condiz com o conteúdo do arquivo
ERA000018	O arquivo enviado excede o tamanho máximo permitido
ERA000019	O parâmetro digital referente ao tamanho máximo permitido deve ser configurado
ERA000020	O arquivo enviado possui caracteres inválidos para a codificação ISO 8859 1
ERA000021	O nome do arquivo deve possuir extensão XML
ERA000022	O parâmetro digital referente ao tamanho máximo permitido não possui valor no formato adequado
ERA000023	O parâmetro digital referente ao tamanho máximo permitido não possui uma unidade de medida conhecida
ERA000024	Transação não encontrada para o estabelecimento informado

```
{
  "erros": [
    {
      "codigo": "ERAXXXXXX",
      "detalhe": "Informações úteis sobre o erro"
    }
  ]
}
```

Exemplo http status code: 460

## Orientações e insumos para os testes

### ORIENTAÇÕES

- O ambiente de homologação é um ambiente que foi preparado/moldado para realização dos testes de modo que suas informações não refletem os dados do ambiente de produção;
- Para realização dos testes, algumas informações são fixas, ou seja, não podem ser alteradas nas chamadas dos testes de serviço. Ex.: "codigoEstabelecimento";
- As demais informações como "codigoAutorização" podem ser editadas nas chamadas dos testes de serviço, mas somente se estiverem listadas na lista de opções abaixo (item 3);
- Para testes com "codigoEstabelecimento" reais, somente é possível no ambiente de produção;

- Para teste de envio de NF em homologação, pode ser enviada uma NF diferente dos dados de transação acima, pois nesse momento do envio não será realizada validação das informações da nota, somente no ambiente de produção essa etapa ocorre;
- Para transações via POS recomendamos armazenar o número de transação/autorização que é impresso no comprovante do pagamento com o cartão, geralmente essa informação é imputada no sistema do posto no momento de dar baixa no cupom, caso o posto não faça esse processo será necessário fazer para garantir a efetividade do processo;
- Caso não se tenha a informação do número de transação/autorização é possível utilizar o serviço de buscar autorização, no entanto, ele vai estar restrito aos parâmetros de tolerância abaixo e caso a informação do cupom estiver fora desses parâmetros abaixo a API vai retornar que não localizou a transação;

### Parâmetros de tolerância da Pesquisa de Transações

Tolerância no Valor da Transação (centavos)

Tolerância no Horário da Transação (minutos)

- Definição número transação: o número de transação é composto por 9 dígitos. Ex.: 999999999;
- Definição número autorização: o número de autorização é composto por 6 dígitos. Ex.: 666666;
- Quando utilizados os serviços de buscar dados de emissão e envio de NF com o número de transação (9 dígitos) a API localiza a transação através dessa informação sem a necessidade de validar pela data;

- Quando utilizados os serviços de buscar dados de emissão e envio de NF com o número de autorização (6 dígitos) a API localiza a transação através dessa informação e a informação da data enviada e ela pesquisa pela transação olhando a data informada + 1 dia anterior a data informada;
- Após envio da NF ela é submetida a uma validação de integridade do arquivo xml, caso o xml esteja integro e legível ele é armazenado em nosso banco dados onde aguardará a geração do pedido correspondente ao período de corte do cliente para ser vinculado ao pedido e passar pelas demais validações das informações que compõe a NF (dados emissor, dados destinatário, valores, etc..). Caso o xml seja reprovado nessa validação inicial o posto receberá um email automático com essa informação, nesse caso deve-se ser verificado o que houve, ajustar e reenviar a NF novamente;
- A API do RAU, por padrão, aceita NFs apenas de transações cujos pedidos tenham sido gerados nos últimos 30 dias ou transações que ainda não tenham pedido gerado, por isso recomendamos sempre o envio da NF logo após o abastecimento (não é necessário esperar a geração do pedido para envio d NF, elas podem ser enviada via API/Integração logo após a realização do abastecimento). Para transações de pedidos mais antigos é possível realizar o envio, mas para que essas NFs sejam integradas ao sistema o posto deve entrar em contato com o Setor Simplifica Fiscal que é responsável pelo processo, informar qual foi o período das transações e pedidos para que a integração das NFS seja realizada;



- Após envio de NF via API/Integração não é necessário realizar o upload no portal do estabelecimento, pois quando os pedidos para essas notas forem gerados o nosso sistema vincula automaticamente as NFs aos pedidos correspondentes conforme o número de transação e, se todas as NFs foram enviadas, os pedidos serão gerados e concluídos de forma automática sem a necessidade de intervenção no portal. Caso alguma NF seja reprovada por dados incorretos será necessário corrigir a informação da NF e a nova NF poderá ser enviada tanto via portal como via API/Integração;

- **Se não for encontrado dados para emissão da NF-e, o que fazer?**

Existem três possíveis cenários:

1. Clientes que possuem o serviço de recolha de notas através da Ticketlog ou Expers: o sistema irá retornar os dados para emissão; (necessário emitir NF e realizar o envio via API)
2. Clientes que não possuem o serviço de recolha de notas através da Ticketlog ou Expers: sistema retorna que o cliente não possui o serviço de recolha de notas; (não é necessário enviar a NF para a Ticket Log/Expers, nem via portal nem via API)
3. Transações não localizadas: esse cenário somente ocorre quando a rede não possui o código de autorização atrelado ao cupom, isso diminui a assertividade da identificação das transações e algumas delas podem acabar não sendo identificadas, isso requer tratativa manual para estes casos em específico (Upload de NF deve ser realizado manual via portal). Além disso por ser que o posto classifique equivocadamente algumas transações Goodcard como Ticketlog ou Expers e ao consultar via API vai retornar não localizadas também, para esses casos não é necessário nenhum procedimento pois esses abastecimentos não possuem o serviço de recolha de notas pela Ticketlog/Expers;

- Para identificação do estabelecimento recomendamos utilizar o atributo "codigoEstabelecimento", todo estabelecimento pode possuir 1 ou 2 códigos de estabelecimento de modo que cada código representa um cadastro/credenciamento diferente, listamos abaixo alguns possíveis cenários:
- 1 código de estabelecimento: 1 código referente ao cadastro Ticketlog
- 1 código de estabelecimento: 1 código referente ao cadastro Expers
- 2 códigos de estabelecimento: 1 código referente ao cadastro Ticketlog e 1 código referente ao cadastro Expers

É importante se atentar a esse ponto pois o mesmo estabelecimento (mesmo cnpj do posto) pode ter os 2 cadastros de credenciamento conosco, que representam produtos e cartões diferentes (Produto Ticketlog e Produto Expers) e cada um deles tem um cadastro separado.

## Insumos para os testes

### 1. AMOSTRA DE TRANSAÇÕES EM HOMOLOGAÇÃO

#### Exemplos de Transações

codigoAutorizacao	dataHoraCupom	valorCupom	VolumeAbastecido	numeroAutorizacao
397380127	26/02/2020 17:40	R\$ 504,00	30,83	380127
397380128	26/02/2020 17:40	R\$ 207,00	15,00	380128
397380327	24/03/2020 09:25	R\$ 2.500,00	30,183	380327
397379829	24/03/2020 16:25	R\$ 2.750,00	30,83	379829
397429311	24/03/2020 18:25	R\$ 2.890,00	30,83	429311
397380129	25/03/2020 10:56	R\$ 504,00	30,83	380129
397380130	25/03/2020 10:56	R\$ 207,00	15,00	380130

### 2. BUSCA DADOS NFE

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** `https://serviceshml.ticketlog.com.br/ticketlog-servicos/credenciamento/recolhaAutonoma/dadosEmissaoNfeRecolhaAutonoma?codigoEstabelecimento=4176677&codigoAutorizacao=397380127&dataAutorizacao=2020-02-26 00:00:00'`
- Params:** Auth
- Body:**

```
{ 1 2   "cnpj": "62355277000194", 3   "razaoSocial": "INDUSTRIA CONSTRUCOES E MONTAGENS INGELEC S.A. - INCOMISA", 4   "inscricaoEstadual": "528.137.838.11", 5   "endereco": { 6     "logradouro": "AV JULIO DE PAULA CLARO", 7     "numero": "1280", 8     "complemento": "", 9     "bairro": "FEITAL"  }
```
- Status:** 200 OK
- Time:** 1187 ms
- Size:** 898 B

## 3. UPLOAD NFE



