

# Cloud Printing Made Simple

SendFileToPrinter – Cloud-based Label Printing API

Steven Si

*Sr. Software Engineer*



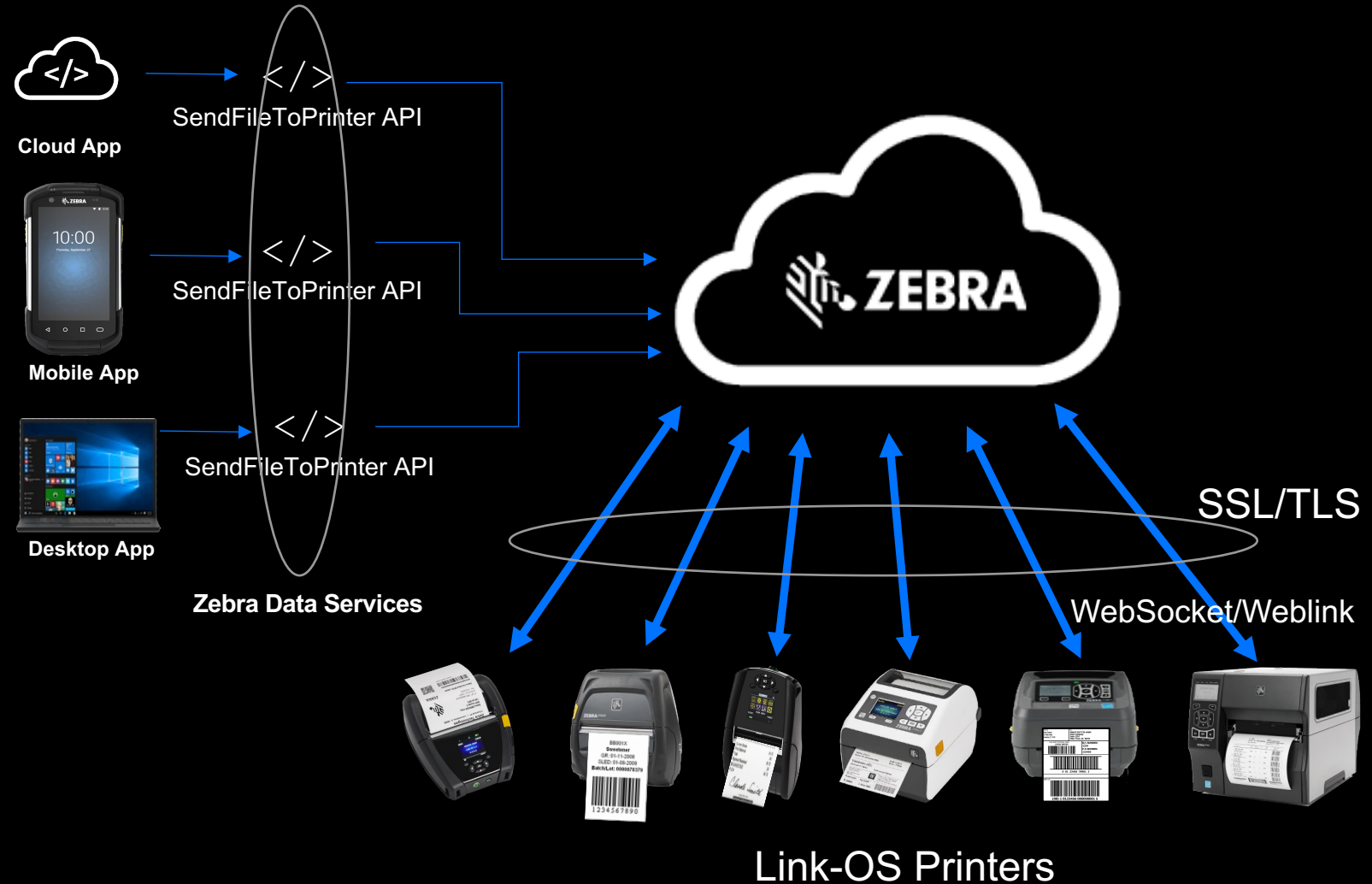
# Agenda

- What is the SendFileToPrinter API?
- What can the API do?
- What does the API look like?
- How to Get Started?
- How to Enroll a Printer?
- How to Use the API?
- Print from Cloud Demo

# What is the SendFileToPrinter API?

## A unidirectional REST API

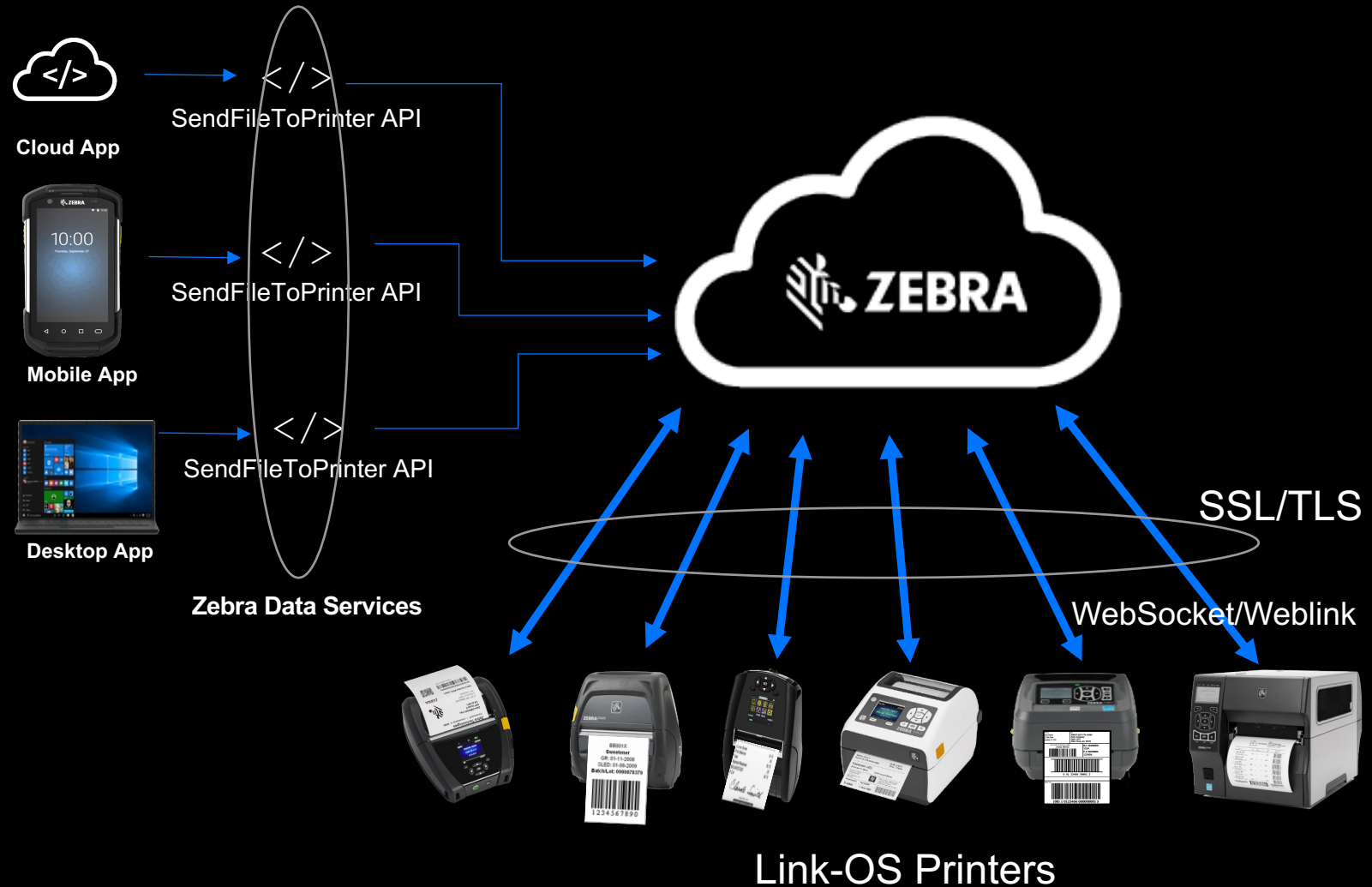
- A REST API
  - Language & platform Agnostic
  - Any programming language on any platform
- Secured Connection
  - SSL/TLS between the printer & Zebra Cloud
  - Encrypted & secured
- Unidirectional
  - No Get/Retrieve from the printer



# What can the SendFileToPrinter API do?

Send any content defined in [ZPL Programming Guide](#)

- Send any content defined in [ZPL Programming Guide](#)
  - ZPL Formats
  - Fonts, Graphics, WiFi Certificates or Objects
  - Configuration Commands
    - Set-Get-Do (SGD, no GET)
    - JSON
- Send up to 10MB content in a file
- Print from Anywhere
  - Cloud App
  - Mobile App
  - Desktop App
- Any Link-OS printers



# What does the API look like?

<https://developer.zebra.com/apis/sendfiletoprinter>

- A multipart/form-data HTTPS request
- Header:
  - API Key (apikey)
  - Tenant account number (Tenant ID)
- Request body
  - One or more printer serial numbers (sn)
  - One file to be sent to the printer (zpl\_file)
- Request URL
  - <https://api.zebra.com/v2/devices/printers/send>

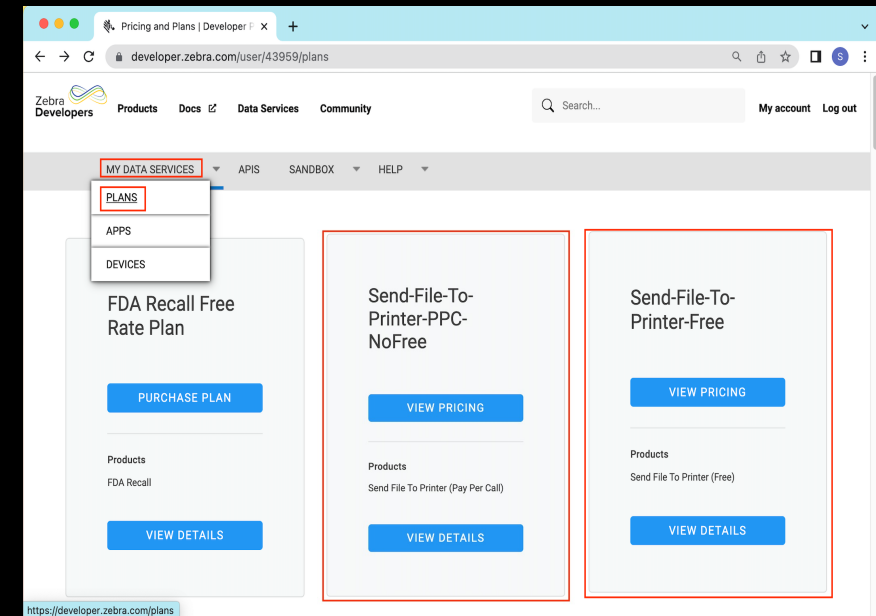
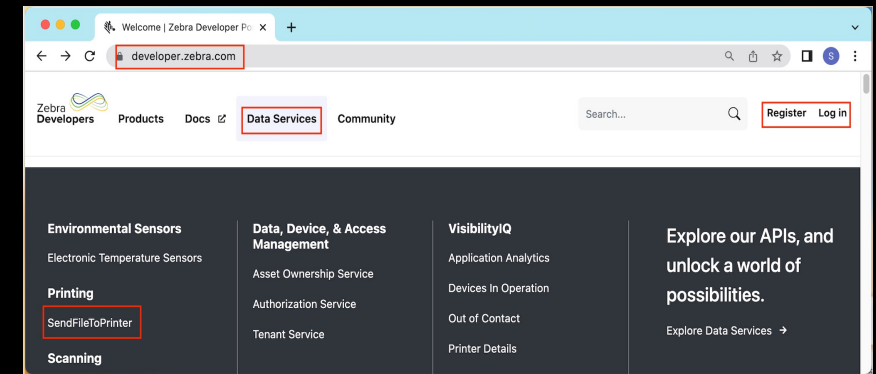
```
$ curl -H "apikey: yourApiKey" -H "tenant: yourTenantAccountNumber" \  
-F "sn=printerSerialNumber_1" \  
-F "sn=printerSerialNumber_2" -F "zpl_file=@HelloWorld.txt" \  
https://api.zebra.com/v2/devices/printers/send
```

# How to Get Started?

Create an account & add the SendFileToPrinter plan(s)



- Apply for a developer account at <https://developer.zebra.com>
- Follow the instructions in Getting Started with Zebra Data Services guide to
  - Select the SendFileToPrinter (Free, PPC, or both) plan(s)
  - Add the plan(s) to your account
- Free Plan
  - 100 calls per day per API key
- Pay Per Call (PPC) Plan
  - Prepaid
  - \$0.01 per call (against the account balance)



# How to Get Started? – cont.

## Verify the plan(s) added successfully

- Verify the added plan(s)
  - Click on My Account -> Balance and plans
  - Verify the plans listed under Active & Future Purchased Plans section

The screenshot shows the 'Purchased plans' page in the Zebra Developers portal. The 'My account' link is highlighted in the top right. Below it, the 'Balance and plans' tab is selected. The 'Active and Future Purchased Plans' section shows two active plans:

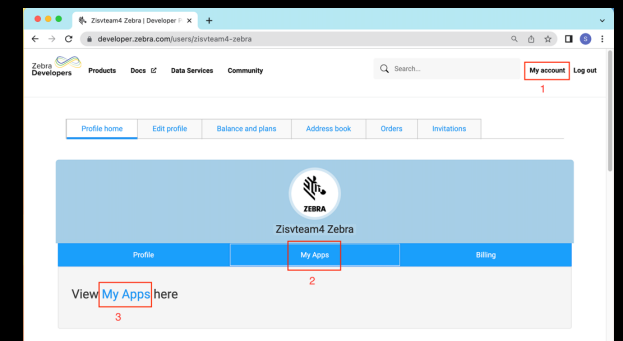
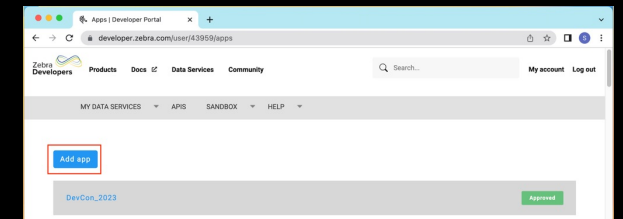
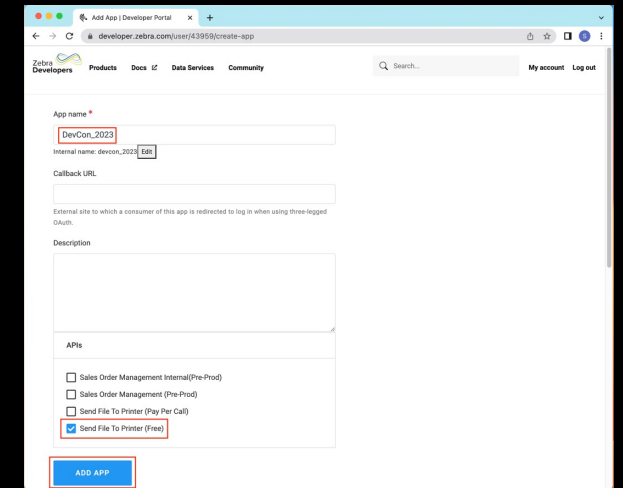
Rate plan	Status	Start Date	End Date	Actions
Send-File-To-Printer-PPC-NoFree	Active	07/20/2020		Cancel
Send-File-To-Printer-Free	Active	07/20/2020		Cancel

Below this section, there is a heading for 'Cancelled and Expired Purchased Plans' with a table structure similar to the one above.

# How to Get Started? – cont.

## Create an app

- Create an app
  - Click on My Account -> My Apps -> My Apps
  - Click on Add app
- Give the app a name
- Check-mark only one of the “Send File To Printer” plans under APIs list
  - Do not choose both PPC and Free plans, otherwise both plans get deducted from the API calls.
- Click on ADD APP button





# How to Get Started? – cont.

## Get the API Key

- Click on My Account -> My Apps -> My Apps

Direct link:

<https://developer.zebra.com/user/me/apps>

- Open the app
- The Consumer Key is the API Key
  - Copy & paste this API key into the REST request header

The screenshot shows the Zebra Developer Portal interface. The browser address bar displays 'developer.zebra.com/user/43959/apps'. The page title is 'Apps | Developer Portal'. A blue 'Add app' button is visible at the top left. The main content area shows an application named 'DevCon\_2023' with a green 'Approved' status. Below the application name are tabs for 'Credentials', 'Edit', 'Delete', and 'Analytics'. The 'Credentials' tab is active, showing a table with the following details:

Details	
Created	2 seconds ago
Last updated	2 seconds ago

Below the details is a 'Credentials' section with a table:

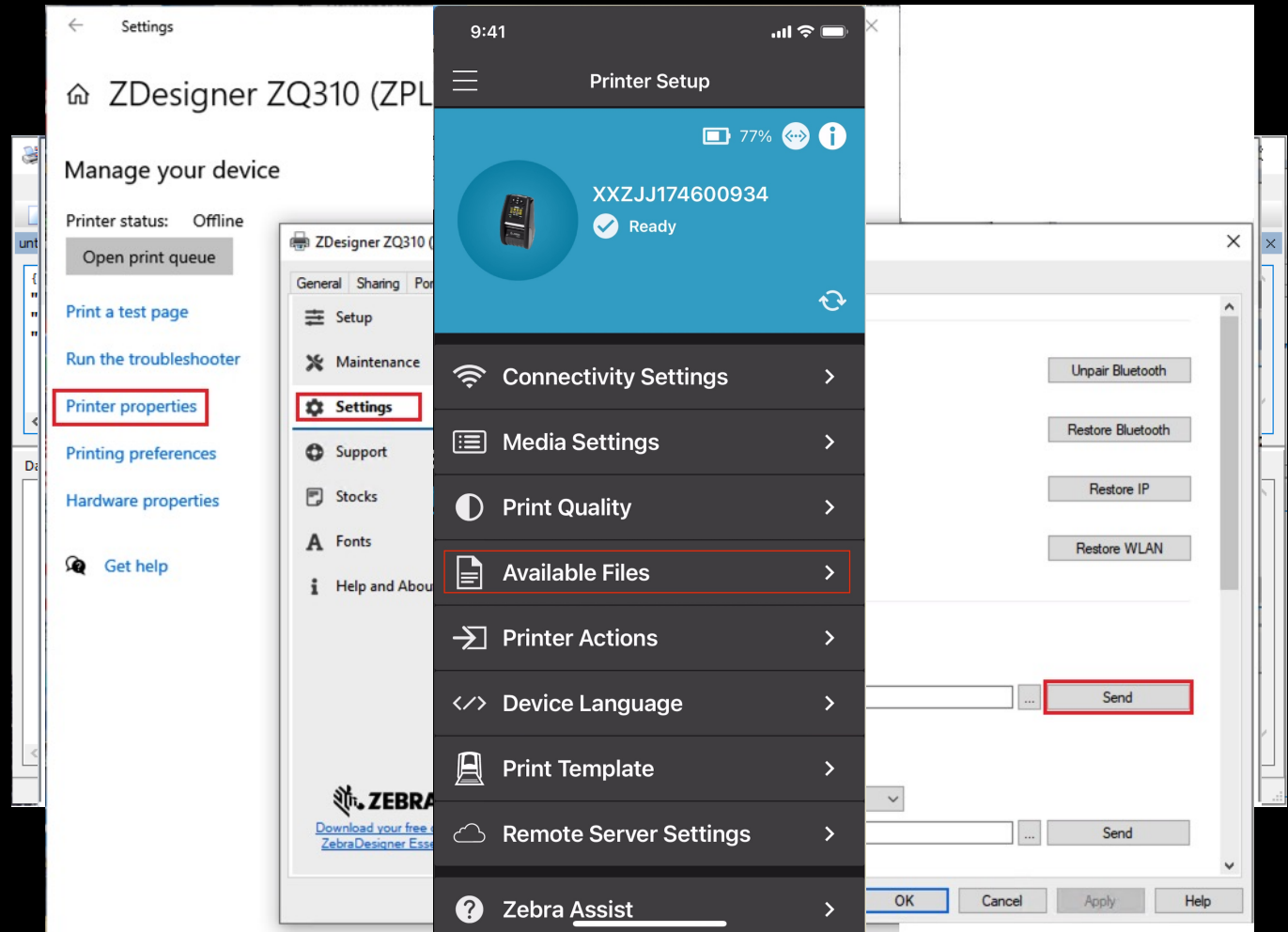
Credentials	
Consumer Key	9qzZiYWUD0QzUKET0ARW L0MQQATuWYq0
Consumer Secret	.....
Issued	1 second ago
Expires	Never
Key Status	Approved

A red box highlights the 'Consumer Key' value, and a red arrow points from the text 'API Key' to this box. To the right of the 'Consumer Key' is a 'Products' section with a toggle for 'Send File To Printer (Free)' which is 'Enabled'.

# How to Enroll a Printer? – cont.

## Send enrollment string to the printer

- Use the send or send file function in Printer Setup Utility for Windows
- Or, use the ZDesigner driver
- Or, use the Printer Setup Utility for Android or iOS



# How to Enroll a Printer?

## Register the printer with Zebra Cloud

- Obtain Printer Enrollment String
- Click on Data Services -> SendFileToPrinter
- Click on MY DATA SERVICES -> DEVICES
- Direct link: <https://developer.zebra.com/my-devices>
- Click on Add Device button
- Select Printer on the dropdown
- Click on Get Enrollment String
- Note:
  1. Tenant ID can be retrieved via Tenant Service
  2. Refer to Printer Setup Guide for details

Add Device - Printer | Developer x +

developer.zebra.com/devices/printer/add

Zebra Developers

Select a Device Type \*

Printer

Tenant \*

zisvteam4@gmail.com

GET ENROLLMENT STRING

Enrolling a device means that that device is accessible via cloud APIs and web-hooks (where available). You can directly communicate with them via APIs, managed services, and support. Enrolling a device includes answering a few basic questions and getting an enrollment code that must be entered on the device. The Add Device button below will provide you with this code.

### Enrollment String:

```
{ "weblink.logging.max_entries": "500", "weblink.ip.conn1.location": "https://savanna-device.zpc.zebra.com/weblink/connect?r=695315b271dd374c76fad074e6b0f9cf", "device.reset": "1" }
```

Tenant

All tenants

Apply

Type	Device ID	Remove
printer	40J13390002	Remove

The r value is the Tenant ID

# How to Enroll a Printer? – cont.

## Verify the Weblink connection

- Retrieve the Weblink log with the SGD  
! U1 getvar "weblink" (Provided a newline at the end)
- Watch for the following messages in the log for a successful enrollment.
  - [conn1.nn] Successfully connected
  - or
  - [conn2.nn] Successfully connected
- If any error, check the following:
  - Is there any typo in the enrollment string?
  - Is there any whitespaces in the weblink URL?
  - Is the data/time on the printer current?

```
Direct Communication - ZDesigner ZQ610 (ZPL) [ZPL]
File Edit Window Help
untitled1 *
! U1 getvar "weblink"
Provide a newline before clicking on the Send To Printer button above.

Data received from the printer is shown in the window below.
[07-17-2023 18:12:44.658][Info][80020001][conn1.1] SSLv3, TLS change cipher, Change cipher spec
[07-17-2023 18:12:44.664][Info][80020014][conn1.1] SSLv3, TLS handshake, Finished
[07-17-2023 18:12:44.684][Info][80020001][conn1.1] SSLv3, TLS change cipher, Change cipher spec
[07-17-2023 18:12:44.689][Info][80020014][conn1.1] SSLv3, TLS handshake, Finished
[07-17-2023 18:12:44.693][Info][00010024][conn1.1] SSL connection using TLSv1.2 / ECDHE-ECDHE-AES128-GCM-SHA256
[07-17-2023 18:12:44.697][Info][0001001C][conn1.1] Server certificate subject: C=US; ST=Illinois; O=Zebra Technologies; OU=Zebra
[07-17-2023 18:12:44.699][Info][0001001D][conn1.1] Server certificate start date: Jun 20 19:45:50 2023 GMT
[07-17-2023 18:12:44.702][Info][0001001E][conn1.1] Server certificate expire date: Jun 19 19:45:50 2026 GMT
[07-17-2023 18:12:44.705][Info][00010013][conn1.1] Server certificate common name: *.zpc.zebra.com (matched)
[07-17-2023 18:12:44.707][Info][0001002E][conn1.1] Server certificate serial number: CC:AB:A0:F0:BA:96:60:14
[07-17-2023 18:12:44.710][Info][0001002F][conn1.1] Server certificate fingerprint: 4F:72:AD:AE:8C:01:B3:01:28:23:DE:18:58:E9:5E:FI
[07-17-2023 18:12:44.713][Info][00010030][conn1.1] Server certificate issuer: C=US; ST=Illinois; O=Zebra Technologies; CN=Weblink
[07-17-2023 18:12:44.716][Info][00010038][conn1.1] Server certificate verify ok
[07-17-2023 18:12:44.739][Info][00001012][conn1.1] HTTP/1.1 101
[07-17-2023 18:12:44.744][Info][00010007][conn1.1] Content Length is 0
[07-17-2023 18:12:44.751][Info][0000101B][conn1.1] Successfully connected

.....
weblink.logging.max_entries : 500 , Choices: 0-10000
weblink.logging.clear
weblink.ip.
weblink.ip.conn1.
weblink.ip.conn1.location : https://savanna-device.zpc.zebra.com/weblink/connect?r=bb62a0cd13e235750402689a723309b0
weblink.ip.conn1.retry_interval : 35 , Choices: 1-600
weblink.ip.conn1.proxy :
weblink.ip.conn1.maximum_simultaneous_connections : 10 , Choices: 1-100
weblink.ip.conn1.authentication.
weblink.ip.conn1.authentication.add
weblink.ip.conn1.authentication.remove
weblink.ip.conn1.authentication.entries :
weblink.ip.conn1.test.
weblink.ip.conn1.test.location : http://www.zebra.com/apps/linktest
weblink.ip.conn1.test.test_on : failure , Choices: off,failure,interval,both
weblink.ip.conn1.test.retry_interval : 900 , Choices: 1-1800
weblink.ip.conn1.num_connections : 1
weblink.ip.conn1.retry_interval_random_max : 120 , Choices: 0-600
weblink.ip.conn2.
weblink.ip.conn2.location :
weblink.ip.conn2.retry_interval : 43 , Choices: 1-600
weblink.ip.conn2.proxy :
weblink.ip.conn2.maximum_simultaneous_connections : 10 , Choices: 1-100
weblink.ip.conn2.authentication.
weblink.ip.conn2.authentication.add
weblink.ip.conn2.authentication.remove
weblink.ip.conn2.authentication.entries :
weblink.ip.conn2.test.
weblink.ip.conn2.test.location : http://www.zebra.com/apps/linktest
weblink.ip.conn2.test.test_on : failure , Choices: off,failure,interval,both
weblink.ip.conn2.test.retry_interval : 900 , Choices: 1-1800
weblink.ip.conn2.num_connections : 0
weblink.ip.conn2.retry_interval_random_max : 120 , Choices: 0-600
weblink.cloud_connect.
weblink.cloud_connect.enable : on , Choices: on,off
weblink.cloud_connect.connect
weblink.cloud_connect.status : NOT ACTIVE
weblink.cloud_connect.connection_key : 0000000000
weblink.cloud_connect.configuration_key : 0000000000
```

# How to Enroll a Printer? – cont.

## Verify the Weblink connection via the My Devices

- Click on Data Services -> MY DATA SERVICES->DEVICES
  - Direct link: <https://developer.zebra.com/my-devices>
- Verify if the printer is listed

My Devices | Developer Portal

developer.zebra.com/my-devices

Zebra Developers Products Docs Data Services Community

MY DATA SERVICES PLANS APPS DEVICES

the backbone of Zebra

and our customer's ability to track assets and communicate between people and those assets. With real-time visibility being more important than ever, these devices are integral to enterprise intelligence.

Below you will find a list of devices that you or your company has claimed as owning. You can also enroll a Zebra fixed RFID reader or Link-OS printer into the service. Enrolling a device means that that device is accessible via cloud APIs and web-hooks (where available). You can directly communicate with them via APIs, managed services, and support. Enrolling a device includes answering a few basic questions and getting an enrollment code that must be entered on the device. The Add Device button below will provide you with this code.

**Requirements**

- See the Printer requirements in [SendFileToPrinter](#).
- See the RFID Fixed Reader requirements in [Data Services for RFID](#).
- [Printer Setup/Enrollment Instructions](#)
- [FX Reader Enrollment Instructions](#)
- Please contact your account manager to access Data Services for RFID.
- If you enrolled a reader using the beta app method, you will need to do an 'enterprise reset' (checkbox in the Shutdown menu on the reader) and then flash to the latest firmware version that is available on our [Support portal](#)
- Network port 443 needs to be open on your network for secure SSL enrollment of printers and RFID readers.

Add Device

Tenant

All tenants

Apply

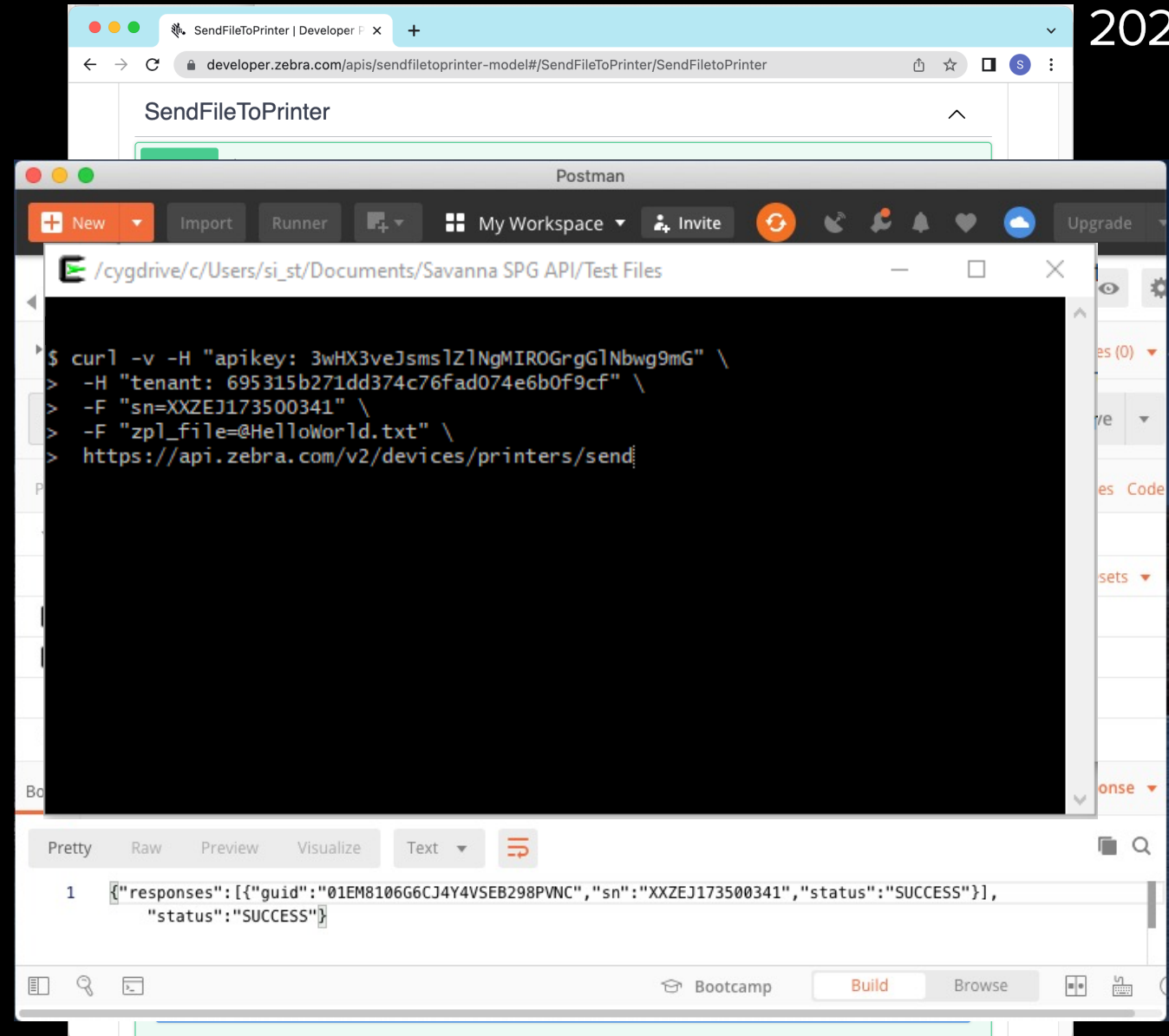
Type	Device ID		Remove
printer	40J133900002	zisvteam4@gmail.com	Remove

<https://developer.zebra.com/my-devices>

# How to Use the SendFileToPrinter API?

## For testing

- [SendFileToPrinter API Portal](https://developer.zebra.com/apis/sendfiletoprinter)
- Direct Link:  
<https://developer.zebra.com/apis/sendfiletoprinter>
- Postman
- cURL



The image shows a browser window displaying the 'SendFileToPrinter' API page on the Zebra Developer Portal. Below it, a Postman window is open, showing a cURL command used to test the API. The cURL command includes headers for 'apikey', 'tenant', and 'sn', and a file path for 'zpl\_file'. The response is a JSON object indicating a successful status.

```
$ curl -v -H "apikey: 3wHX3veJsms1Z1NgMIROGrG1Nbwg9mG" \
> -H "tenant: 695315b271dd374c76fad074e6b0f9cf" \
> -F "sn=XXZEJ173500341" \
> -F "zpl_file=@HelloWorld.txt" \
> https://api.zebra.com/v2/devices/printers/send
```

```
1 [{"responses": [{"guid": "01EM8106G6CJ4Y4VSEB298PVNC", "sn": "XXZEJ173500341", "status": "SUCCESS"}, {"status": "SUCCESS"}]}
```

# How to Use the SendFileToPrinter API? – cont.

## Java with [ZebraSftpMpBodyPublisher.java](#)

```
// Create URI from the URL of SendFileToPrinter API
    uri                "https://api.zebra.com/v2/devices/printers/send"

String sn = "XXZJJ174600974"; // Printer serial number example
    apikey         "EvgdePLp2WMPk8ZTqrAGDAGqCzs83Ajn" // API Key example
    tenant         "695315b272dd374c76fad074e6b0f9cf" // Tenant ID example

// Example ZPL file containing - "^XA ^FO50,50^ADN,36,20^FDHello World^FS ^XZ"
    zplFilePath      "C:\\Users\\john\\HelloWorld.txt"

// Use ZebraSftpMpBodyPublisher.java
ZebraSftpMpBodyPublisher publisher
    = new ZebraSftpMpBodyPublisher().addSn(sn).addSn(sn).addSn(sn).setZplFilePath(zplFilePath);

HttpRequest request = HttpRequest.newBuilder().uri(uri).header("apikey", apikey).header("tenant", tenant)
    .header("Content-Type", "multipart/form-data; boundary=" + publisher.getBoundary())
    .POST(publisher.build()).build();

HttpClient httpClient = HttpClient.newBuilder().build();
HttpResponse response = httpClient.send(request, BodyHandlers.ofString());

System.out.println(response.body());
```

# How to Use the SendFileToPrinter API? – cont.

## JavaScript with [PrintFromCloud demo on GitHub](#)

```
// Call the SendFileToPrinter API to send the file
function sendFileToPrinter(apiKey, tenant, fileToSend, serialNumbers) {

    var http = new XMLHttpRequest();
    http.open("POST", "https://api.zebra.com/v2/devices/printers/send", true);

    // Set the proper header information for the request, including the API Key.
    // Do not specify the Content-Type header here, as it is implied in
    http.setRequestHeader("apikey", apiKey); // Add API key to the header
    http.setRequestHeader("tenant", tenant); // Add tenant ID

    // Call a function when the state changes.
    if((http.readyState == 4 || http.readyState == 1) ) {
        // alert(http.responseText);
    }
}

fd = new FormData();
    "zpl_file" // Attach the file to be sent.

// Append the printer serial numbers
serialNumbers.forEach(function(item, index, array) {
    fd.append("sn", item);
})

http.send(fd);
}
```



# Resources

## Documents & Demo

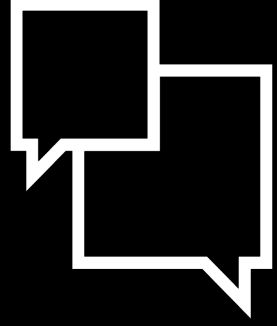


- [Use the SendFileToPrinter API for Your Cloud-Based Printing Needs](https://developer.zebra.com/blog/use-sendfiletoprinter-api-your-cloud-based-printing-needs)  
(<https://developer.zebra.com/blog/use-sendfiletoprinter-api-your-cloud-based-printing-needs>)
- [Getting Started with Zebra Data Services](https://developer.zebra.com/docs/getting-started)  
(<https://developer.zebra.com/docs/getting-started>)
- [Printer Setup Guide](https://developer.zebra.com/docs/printer-setup-sendfiletoprinter)  
(<https://developer.zebra.com/docs/printer-setup-sendfiletoprinter>)
- Print from cloud example in JavaScript on GitHub – [PrintFromCloud](https://github.com/ZebraDevs/Zebra-Printer-Samples/tree/Print-From-Zebra-Data-Services-Cloud-Example)  
(<https://github.com/ZebraDevs/Zebra-Printer-Samples/tree/Print-From-Zebra-Data-Services-Cloud-Example>)

# Print From Cloud Walkthrough

## JavaScript based demo

- Send a file to printer
- Template-based label print – Badge
- Template-based receipt print – Shopping Receipt



# Questions

# Zebra DevCon 2023



# Thank You

ZEBRA and the stylized Zebra head are trademarks of Zebra Technologies Corp., registered in many jurisdictions worldwide. All other trademarks are the property of their respective owners.  
©2023 Zebra Technologies Corp. and/or its affiliates. All rights reserved.

