DevCon 2025

Connect | Learn | Build

June 4 - 6, 2025

Amsterdam, Netherlands

ZEBRA

DevCon 2025

Connect | Learn | Build

Harness Cloud Connectivity: Integrate Zebra Printers with Modern Systems

Bret Anno

Director, Software Engineering

ZEBRA

DevCon 2025

Connect | Learn | Build

Cloud Application Printing

ZEBRA

Cloud Application Printing

Overview

Server Side

- Printer connected to cloud server
- Client is logically tied to printer
- Server triggers print operation to printer

+ Universal printing from any browser

– More involved connection process

Client Side

- Printer is connected to client
- Client controls printing

+ Simple connection

+ Immediate feedback on connection

– Client access to printer not universal across browsers and operating systems

DevCon 2025

Connect | Learn | Build

Client Printing

ZEBRA

Windows and CUPS Drivers

Base Printing Option

- Create a view that is just your label
- Format it to a size equivalent to label in the printer
- Call print function that brings up the print dialog
- Browsers supported: All
- OS support: Windows, Mac, Linux
- Ports: TCP, USB

– Too many clicks and options to print

– No mobile support

– Graphics only

Android Print Service

Base Printing Option

- Create a view that is just your label
- Format it to a size equivalent to label in the printer
- Call print function that brings up the print dialog
- Browsers supported: All
- OS support: Android
- Ports: TCP, BT, USB

– Too many clicks and options to print

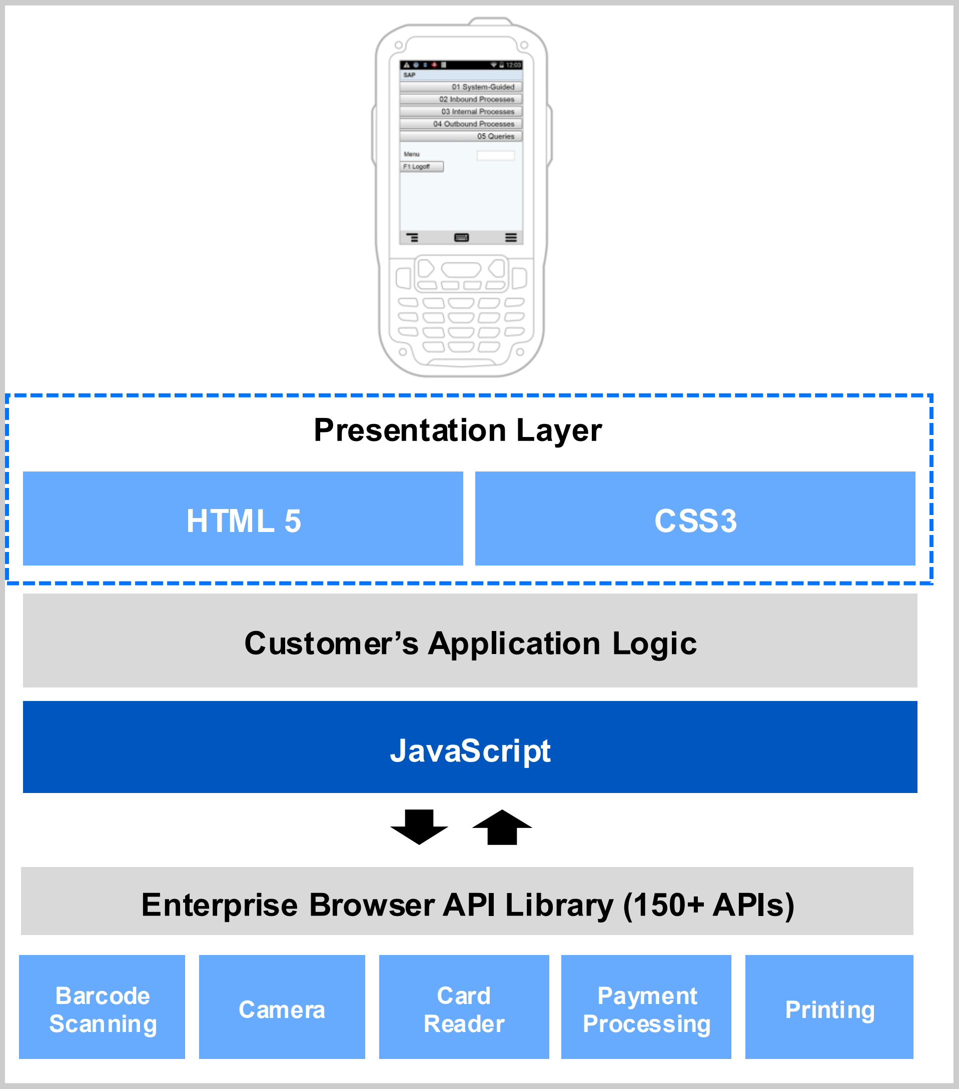
– Only Android

– Graphics only

Enterprise Browser

DevCon 2025

Connect | Learn | Build



Presentation Layer

HTML 5

CSS3

Customer’s Application Logic

JavaScript

Enterprise Browser API Library (150+ APIs)

Barcode Scanning

Camera

Card Reader

Payment Processing

Printing

1 Enterprise Browser is a powerful, industrial browser that provides everything needed to quickly build device apps for **barcode scanning, signature capture, payment processing, printing** and most other enterprise applications.

2 Enterprise Browser is currently supported on Zebra devices running Android or Windows Mobile/CE.

3 EB includes a runtime environment inside which a company's application logic can be executed and controlled using HTML5 and CSS3 for presentation and JavaScript to access EB APIs for scanners, cameras, card readers and other device peripherals.

4 The base EB installation has all of the components that would allow a Windows development host to easily build apps and set runtime settings (via Config.xml file) for local or mass-deployment using Zebra StageNow or a mobile device management (MDM) system.

BrowserPrint

https://developer.zebra.com/products/printers/browser-print

- ZPL passthrough/bi-di
- Graphics printing support
- Status, config, and SGD helper functions
- Leverages loopback in Javascript via localhost to communicate to BrowserPrint application
- Browsers supported: All
- OS support: Windows, MacOS, Android
- Ports: TCP, USB, BT (Android)

– Potential for browsers to stop support even though it is part of the w3c standard(Safari)

```
device.send("^XA^HH^XZ", function(success){
    device.read(function(response){
       console.log(response);
    }, function(error){console.error(error);});
}, function(error){console.error(error);});
```

```
BrowserPrint.convert("https://example.com/myimage.png", device, options, success, failure);
```

```
zebraPrinter.printImageAsLabel(url, function(){}, function(error){});
```

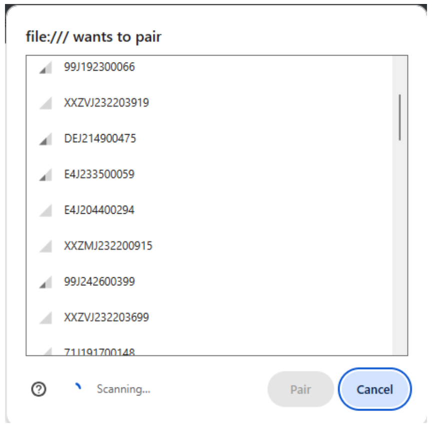
WebBT

- Browser controls selection
- BlueTooth Low Energy

- Browsers supported: Chrome, Opera, Edge
- OS support: Android, Windows, Mac
- Ports: BLE

– Only BT and no iOS

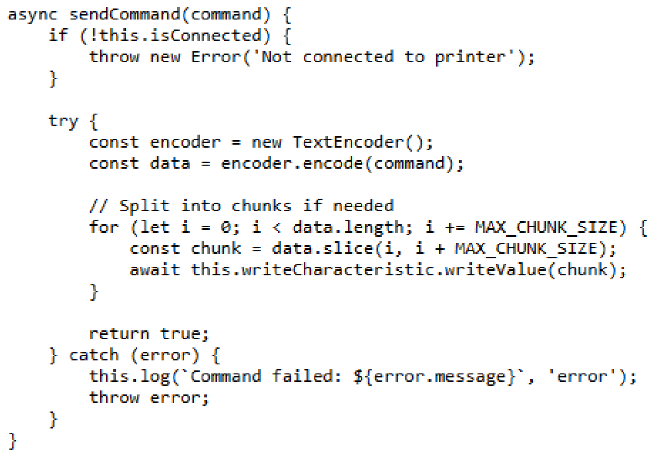
– Performance varies



file:/// wants to pair

99J192300066
XXZVJ232203919
DEJ214900475
E4J233500059
E4J204400294
XXZMJ232200915
99J242600399
XXZVJ232203699
71J191700148

Scanning...

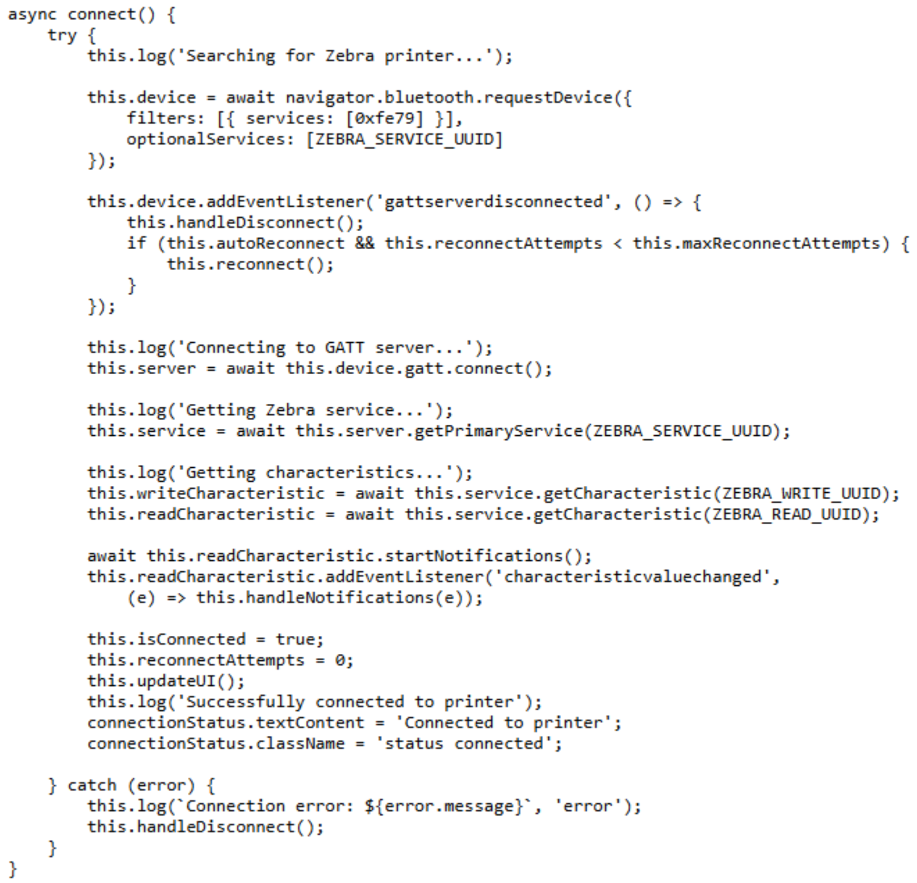


```
async sendCommand(command) {
    if (!this.isConnected) {
        throw new Error('Not connected to printer');
    }

    try {
        const encoder = new TextEncoder();
        const data = encoder.encode(command);

        // Split into chunks if needed
        for (let i = 0; i < data.length; i += MAX_CHUNK_SIZE) {
            const chunk = data.slice(i, i + MAX_CHUNK_SIZE);
            await this.writeCharacteristic.writeValue(chunk);
        }

        return true;
    } catch (error) {
        this.log(`Command failed: ${error.message}`, 'error');
        throw error;
    }
}
```



```
async connect() {
    try {
        this.log('Searching for Zebra printer...');

        this.device = await navigator.bluetooth.requestDevice({
            filters: [{ services: [0xfe79] }],
            optionalServices: [ZEBRA_SERVICE_UUID]
        });

        this.device.addEventListener('gattserverdisconnected', () => {
            this.handleDisconnect();
            if (this.autoReconnect && this.reconnectAttempts < this.maxReconnectAttempts) {
                this.reconnect();
            }
        });

        this.log('Connecting to GATT server...');
        this.server = await this.device.gatt.connect();

        this.log('Getting Zebra service...');
        this.service = await this.server.getPrimaryService(ZEBRA_SERVICE_UUID);

        this.log('Getting characteristics...');
        this.writeCharacteristic = await this.service.getCharacteristic(ZEBRA_WRITE_UUID);
        this.readCharacteristic = await this.service.getCharacteristic(ZEBRA_READ_UUID);

        await this.readCharacteristic.startNotifications();
        this.readCharacteristic.addEventListener('characteristicvaluechanged',
            (e) => this.handleNotifications(e));

        this.isConnected = true;
        this.reconnectAttempts = 0;
        this.updateUI();
        this.log('Successfully connected to printer');
        connectionStatus.textContent = 'Connected to printer';
        connectionStatus.className = 'status connected';

    } catch (error) {
        this.log(`Connection error: ${error.message}`, 'error');
        this.handleDisconnect();
    }
}
```

Hybrid App or Native Client

Logic wrapper

- Full custom control of devices
- Printer SDKs interfaced to control print

DevCon 2025

Connect | Learn | Build

Direct Server Connection

ZEBRA

Overview

Printer connects directly to server

- CloudConnect
  - Print or configuration commands
  - Multiple pipes allowed for status during print
  - Basis of PPME, PPME Cloud, and SendFile2Printer API
  - Open Specification to build your own
  - Secure Websockets with JSON interface

- MQTT
  - Configuration only
  - Basis of Soti Connect
  - Tested against standard brokers (HiveMQ and Mosquitto)

A Printer connected directly to application back end allows front end to exist on any platform

DevCon 2025

Connect | Learn | Build

Cloud Connect

Cloud Connect

Overview

Printing Needs

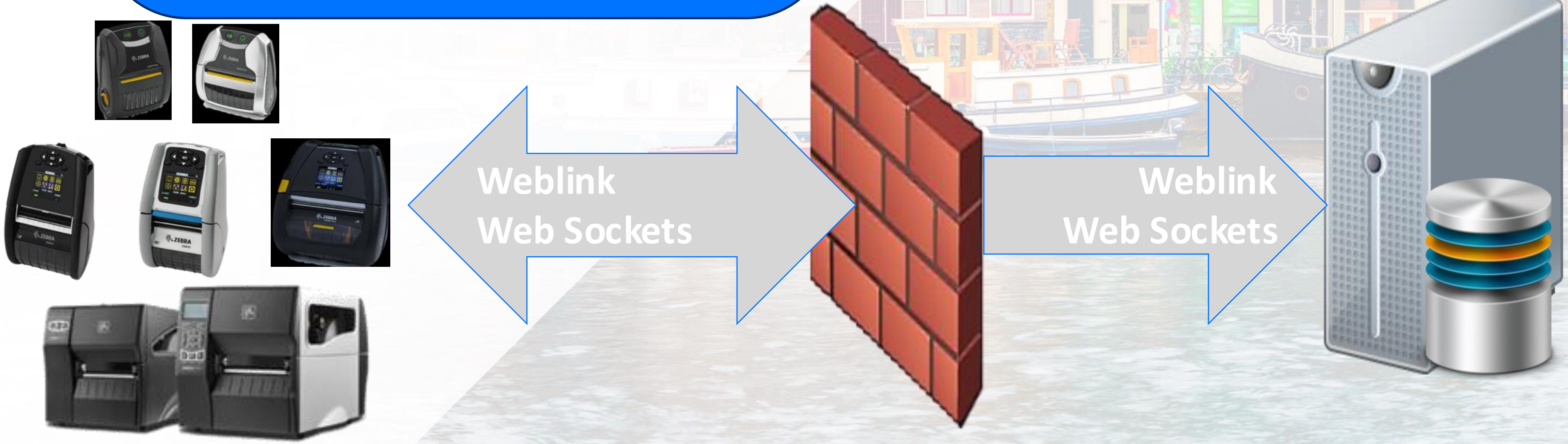
- Low latency downstream
- Firewall Navigation

Websockets

- Secure TLS
- Lower overhead than HTTPS
- Persistent connection
- Low latency
- Need to account for server burden

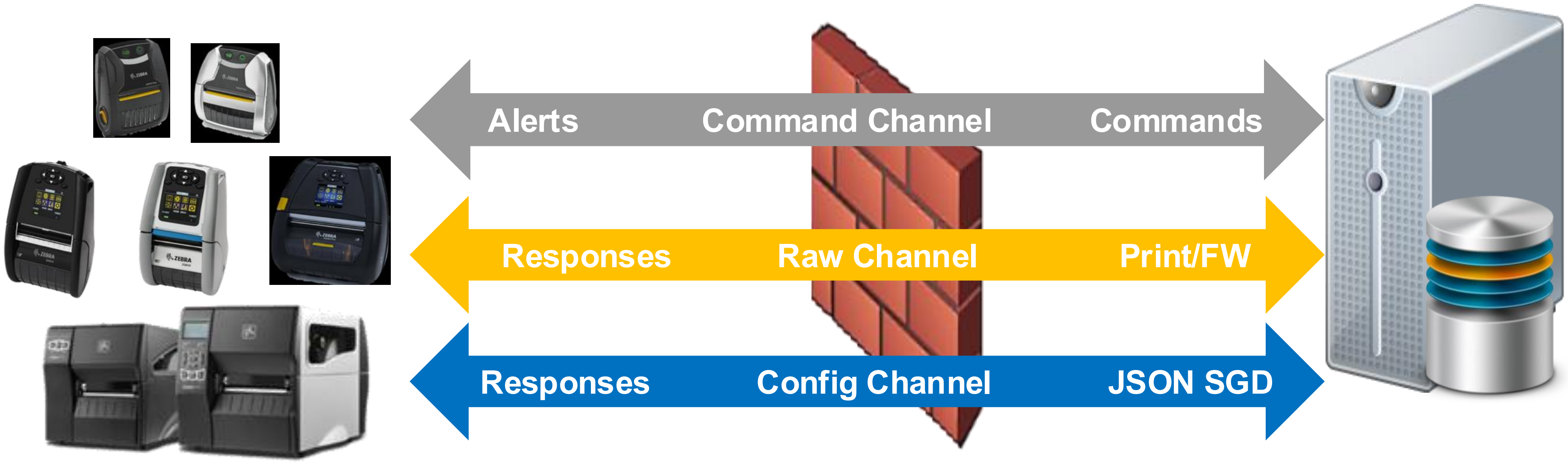
Weblink

- Multi-channel Websockets
- Mature - 10+ years



Cloud Connect

Connections



```
{
  "open" : "v1.raw.zebra.com"
}
```

```
{
  "open" : "v1.config.zebra.com"
}
```

```
{
 "configure_alert" : "HEAD OPEN,SDK,Y,Y,,,N"
}
```

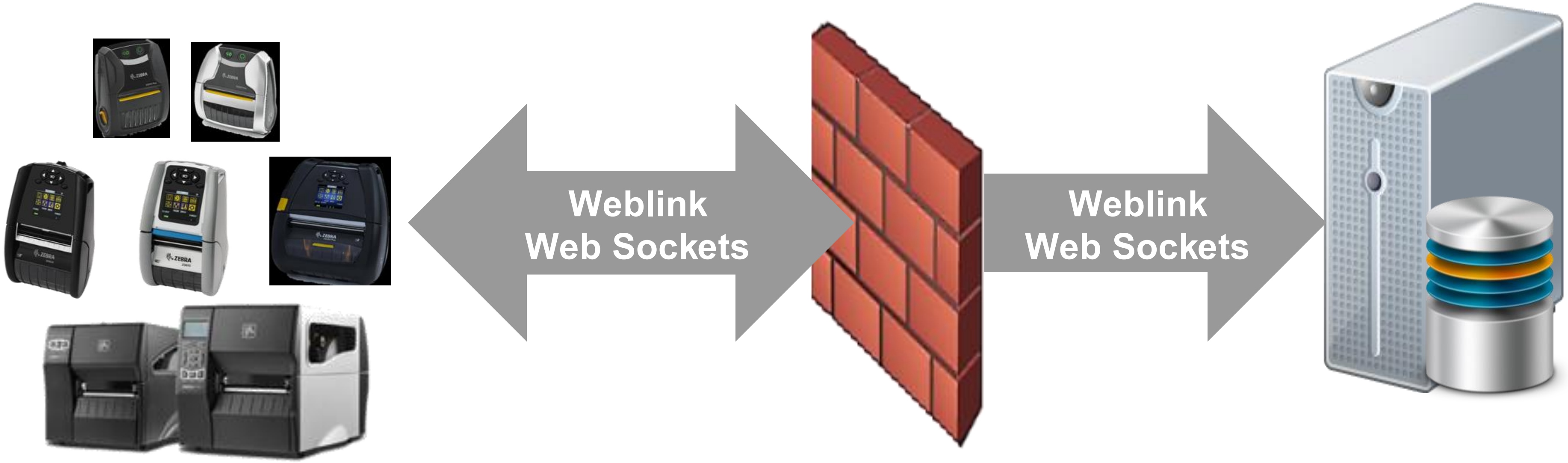
```
{
  "alert" :
    {
        "unique_id" : "XXXYYYZZZ",
        "time_stamp" : "2015-06-09 03:38:12",
        "type_id" : "ERROR",
        "condition_id" : "HEAD OPEN",
        "condition_state" : "SET",
        "type" : "ERROR CONDITION",
        "condition" : "HEAD OPEN"
    }
}
```

Cloud Connect

Connection Overview

DevCon 2025
Connect | Learn | Build

1. Server and printer preloaded with certificates for mutual authentication
2. Printer initiates HTTPS connection to server.
   - Initiate TLS connection
   - Printer requests the server's certificate to verify it is an authorized server.
   - Server requests the printer's certificate to verify it is an authorized printer.
3. Request Upgrade to Web Sockets
4. Server agrees to upgrade & communication is Web Sockets after that.
5. Command channel established and printer sends the server its discovery packet.



Cloud Connect

Certificates

- Zebra CA shutdown
- Self generated certs created unique pairing between printer and server
- PPME will generate or allow certs to be uploaded
- SendFile2Printer (SF2P) uses built in Zebra certificate
  - No printer side certificate loading required

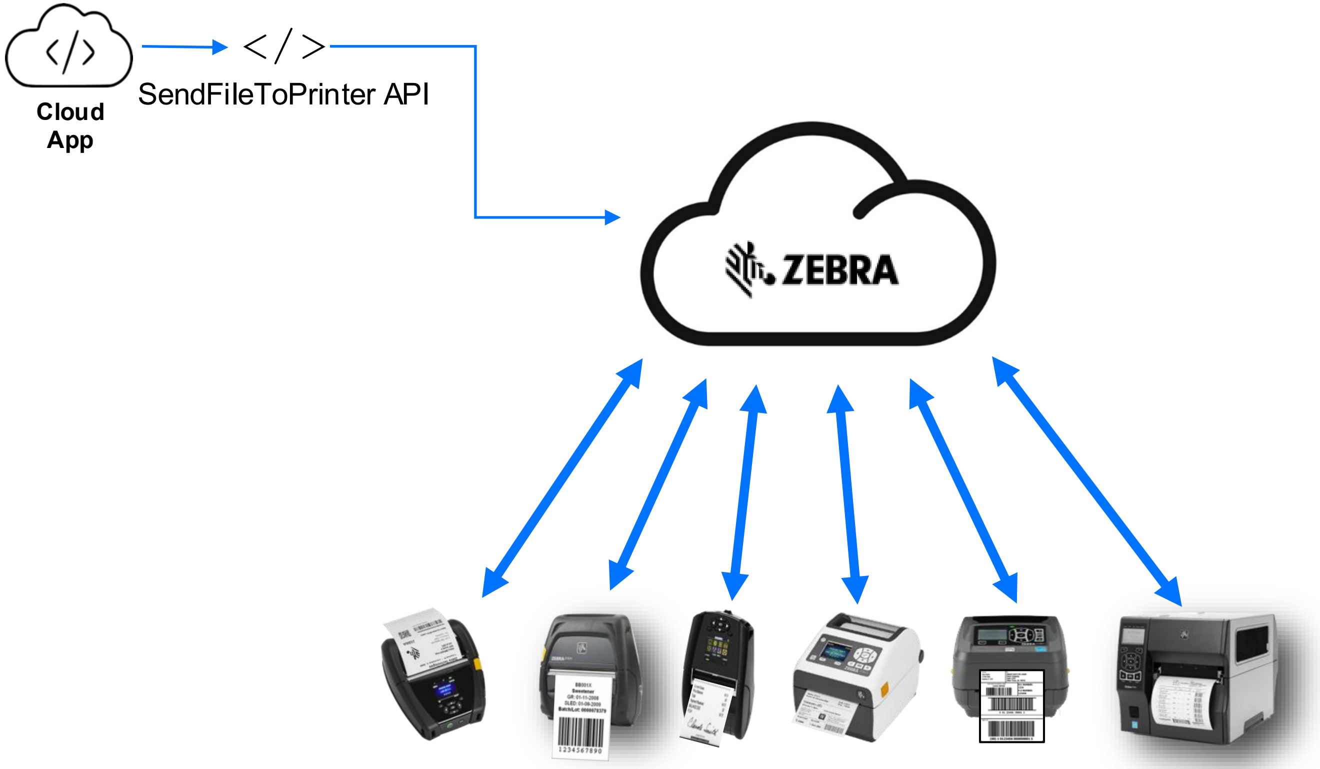
Self Implemented

- Documentation
  - https://www.zebra.com/content/dam/support-dam/en/documentation/unrestricted/application-notes/appnotes-creating-weblink-endpoint-rev1-english.pdf
- Implement using frameworks and tools you normally use
  - Customers have implemented on AWS, GCP, and Azure
  - Language specific SDKs can be used for graphics conversion and other helpful features

SendFileToPrinter SF2P API

DevCon 2025
Connect | Learn | Build

- Zebra hosts printer connections
- Connects raw channel to simple Rest API
  - ZPL, fonts, graphics, config
  - Unidirectional, 10MB limit per call
- Server to Server

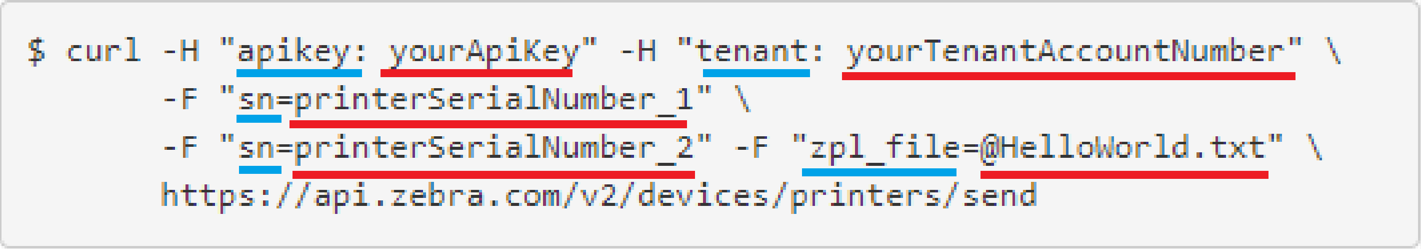


What does the API look like?

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

A multipart/form-data HTTPS request

- Header:
  - API Key (**`apikey`**)
  - Tenant account number (**`Tenant ID`**)



```
$ 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
```

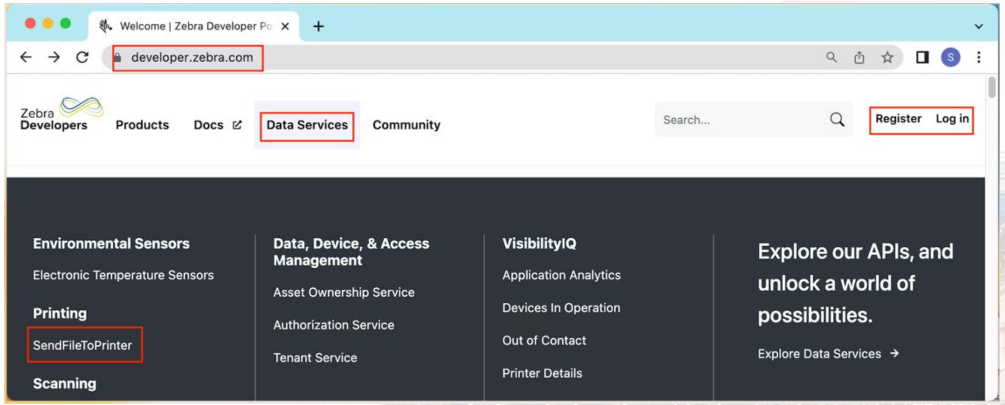
- Request body
  - One or more printer serial numbers (**`sn`**)
  - One file to be sent to the printer (**`zpl_file`**)

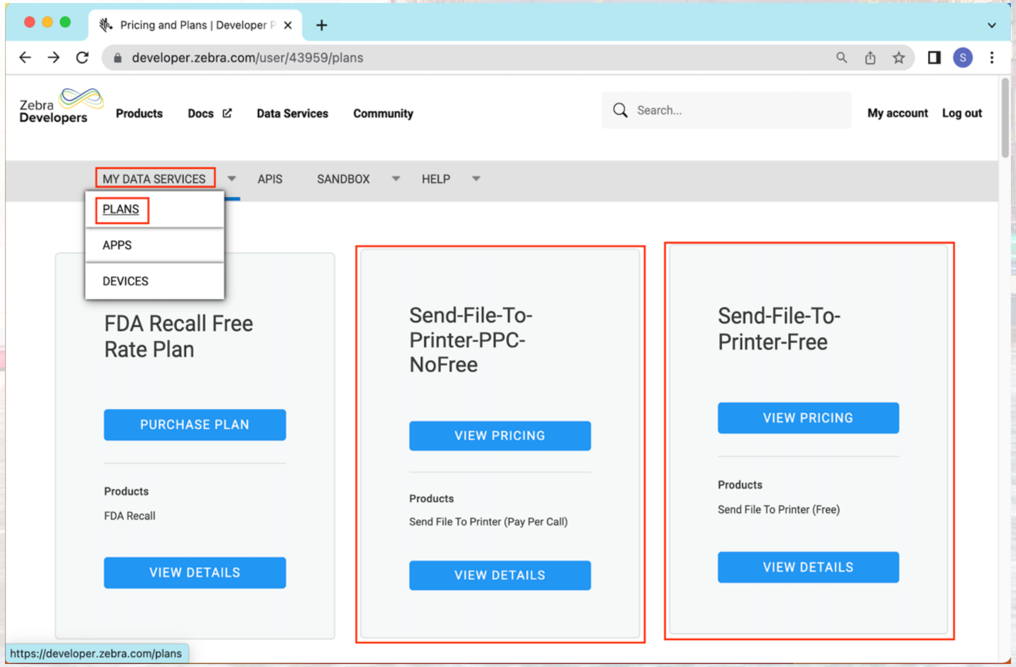
- Request URL
  - `https://api.zebra.com/v2/devices/printers/send`

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)

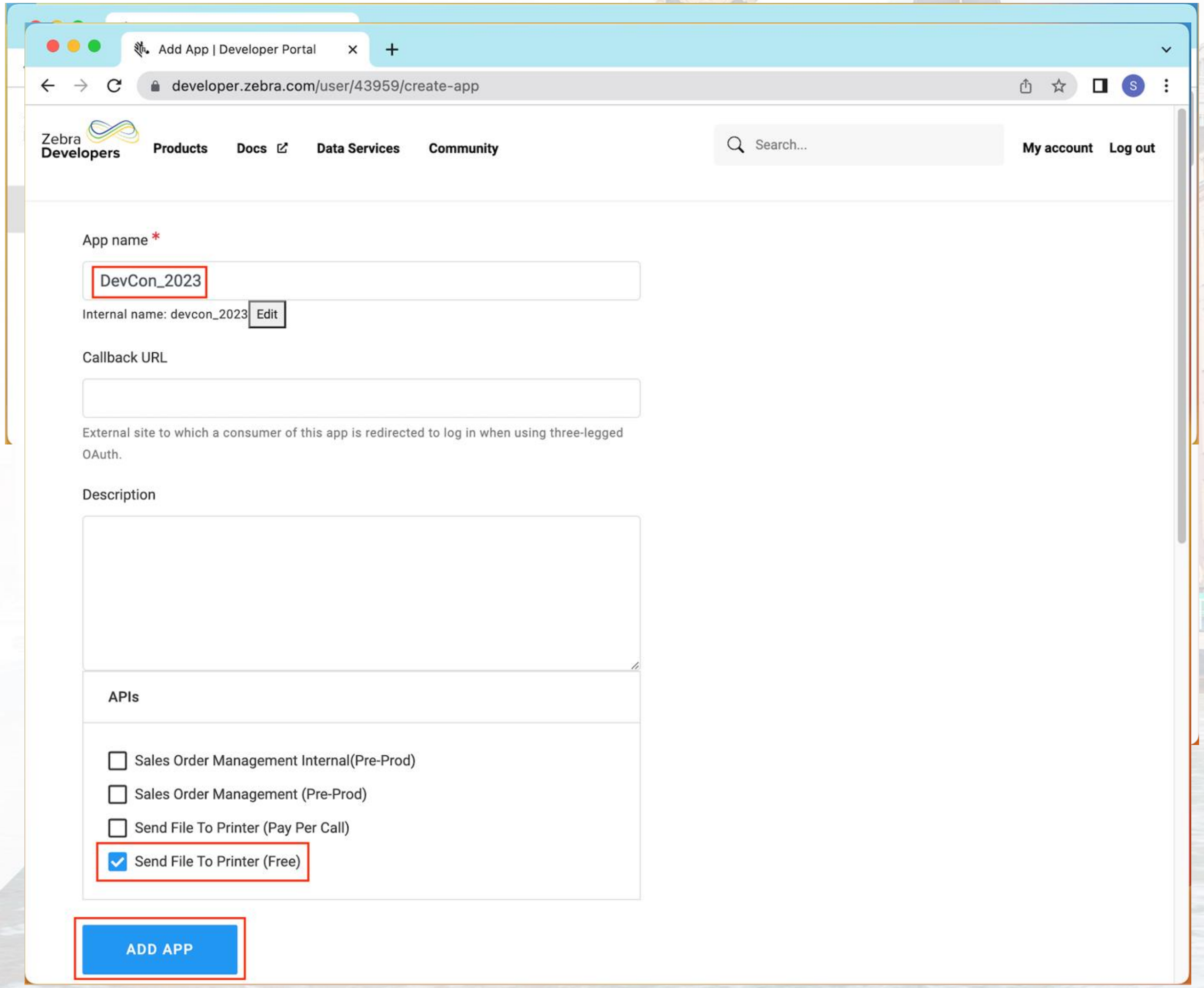



Pricing and Plans | Developer P
developer.zebra.com/user/43959/plans
Zebra Developers
Products
Docs
Data Services
Community
Search...
My account
Log out
MY DATA SERVICES
APIS
SANDBOX
HELP
PLANS
APPS
DEVICES
FDA Recall Free Rate Plan
PURCHASE PLAN
Products
FDA Recall
VIEW DETAILS
Send-File-To-Printer-PPC-NoFree
VIEW PRICING
Products
Send File To Printer (Pay Per Call)
VIEW DETAILS
Send-File-To-Printer-Free
VIEW PRICING
Products
Send File To Printer (Free)
VIEW DETAILS
https://developer.zebra.com/plans

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



App name *

DevCon_2023

Internal name: devcon_2023

Callback URL

External site to which a consumer of this app is redirected to log in when using three-legged OAuth.

Description

APIs

☐ Sales Order Management Internal(Pre-Prod)

☐ Sales Order Management (Pre-Prod)

☐ Send File To Printer (Pay Per Call)

☑ Send File To Printer (Free)

How to Get Started? – cont.

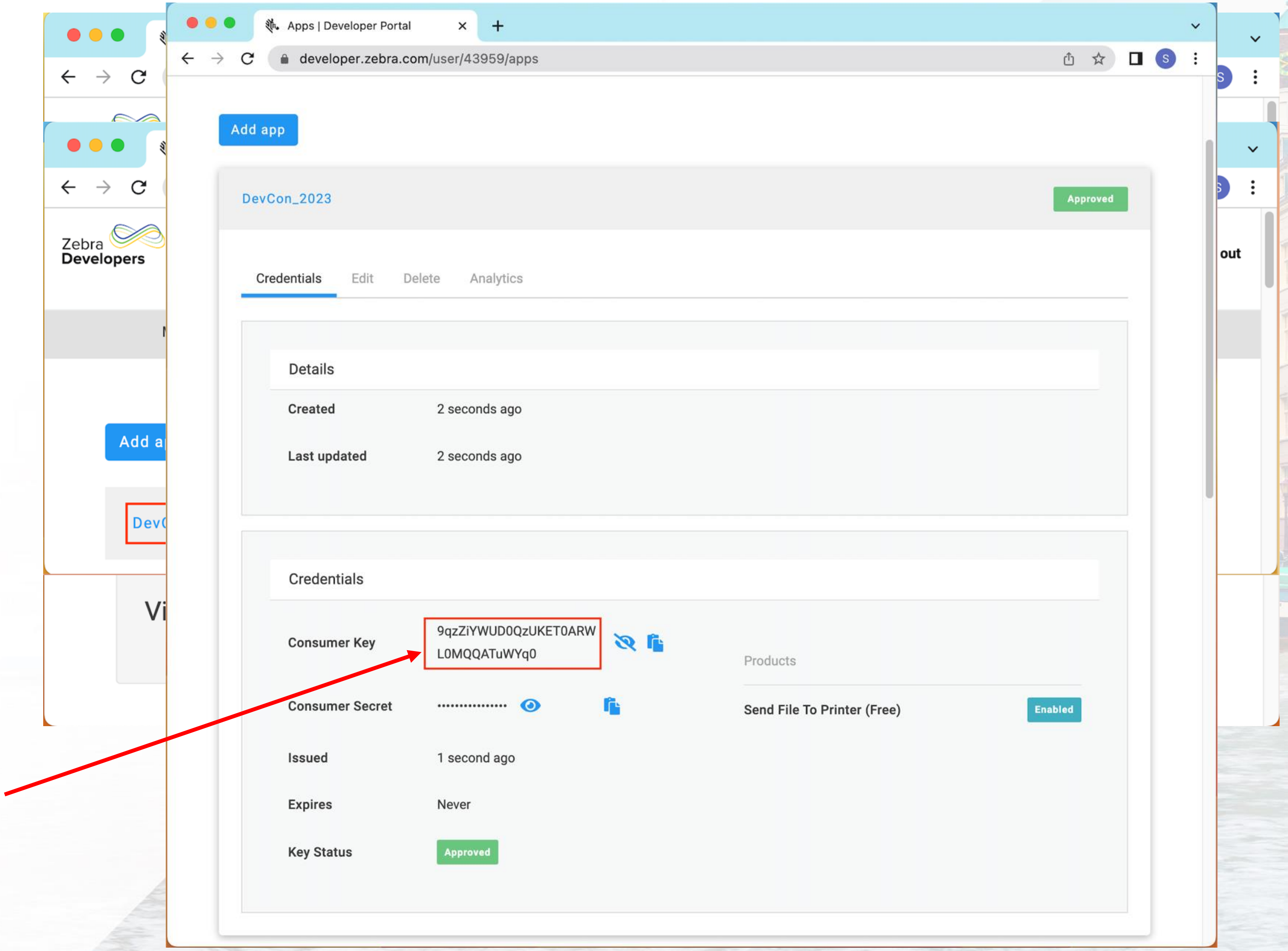
Get the API Key

- 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 the REST request header



DevCon_2023 — Approved

Credentials | Edit | Delete | Analytics

Details

Created: 2 seconds ago

Last updated: 2 seconds ago

Credentials

Consumer Key: 9qzZiYWUD0QzUKET0ARW L0MQQATuWYq0

Consumer Secret: ••••••••••••••••

Issued: 1 second ago

Expires: Never

Key Status: Approved

Products

Send File To Printer (Free) — Enabled

API Key

How to Enroll a Printer?

Register the printer with Zebra Cloud

Obtain Printer Enrollment String

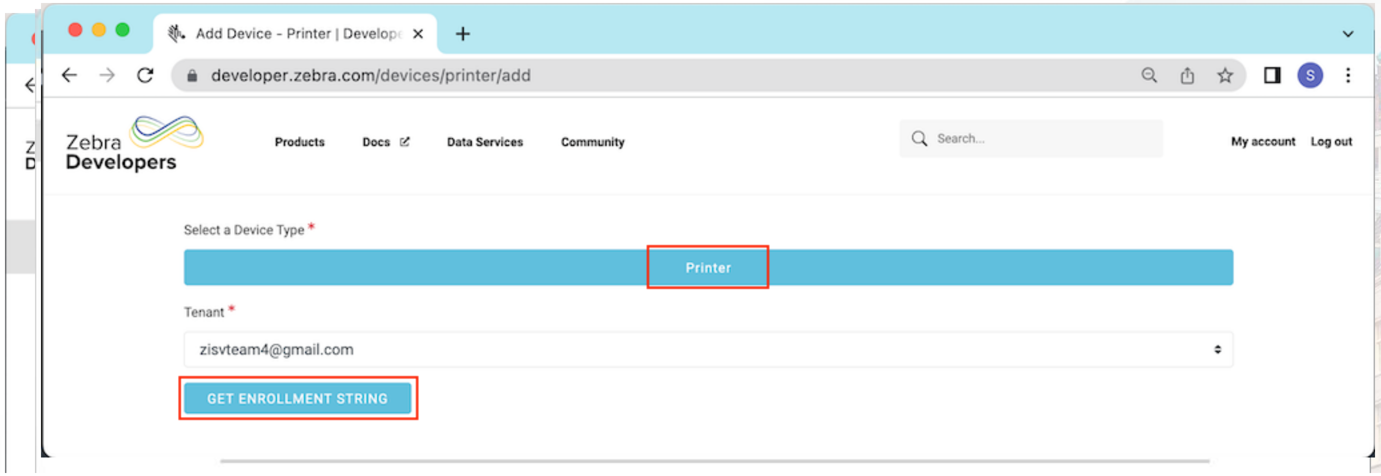
- Click on **Data Services -> SendFileToPrinter**
- Click on **MY DATA SERVICES -> DEVICES**

Direct link: https://developer.zebra.com/my-devices

- Click on **Add Device** button
- Select **Printer** on the dropdown
- Click on **Get Enrollment String**

Note:

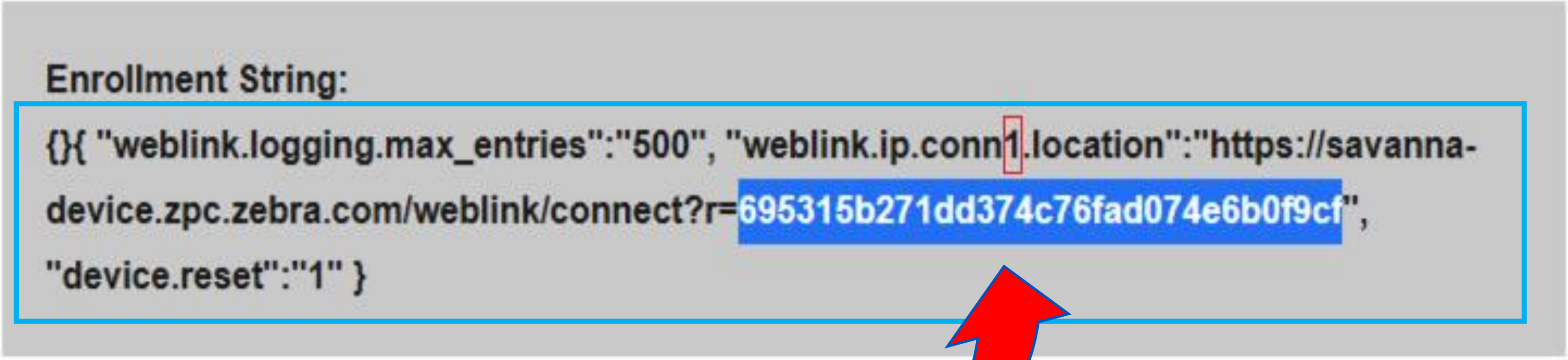
1. Tenant ID can be retrieved via Tenant Service
2. Refer to Printer Setup Guide for details



developer.zebra.com/devices/printer/add

Select a Device Type *: Printer

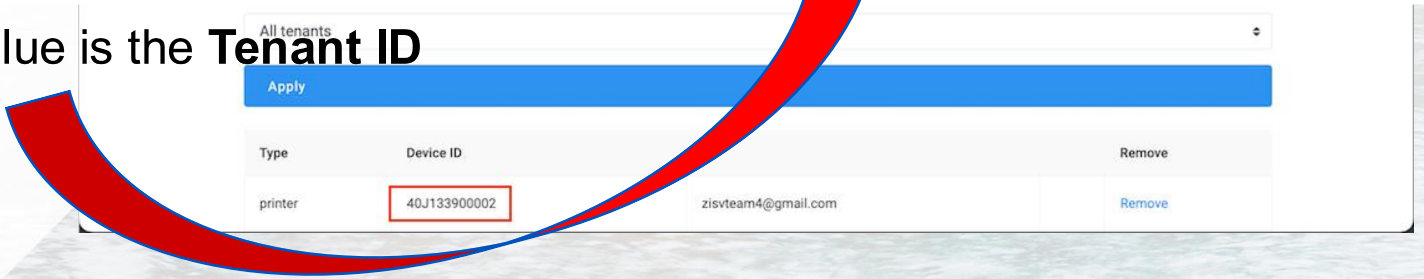
Tenant *: zisvteam4@gmail.com



Enrollment String:

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

The **r** value is the **Tenant ID**

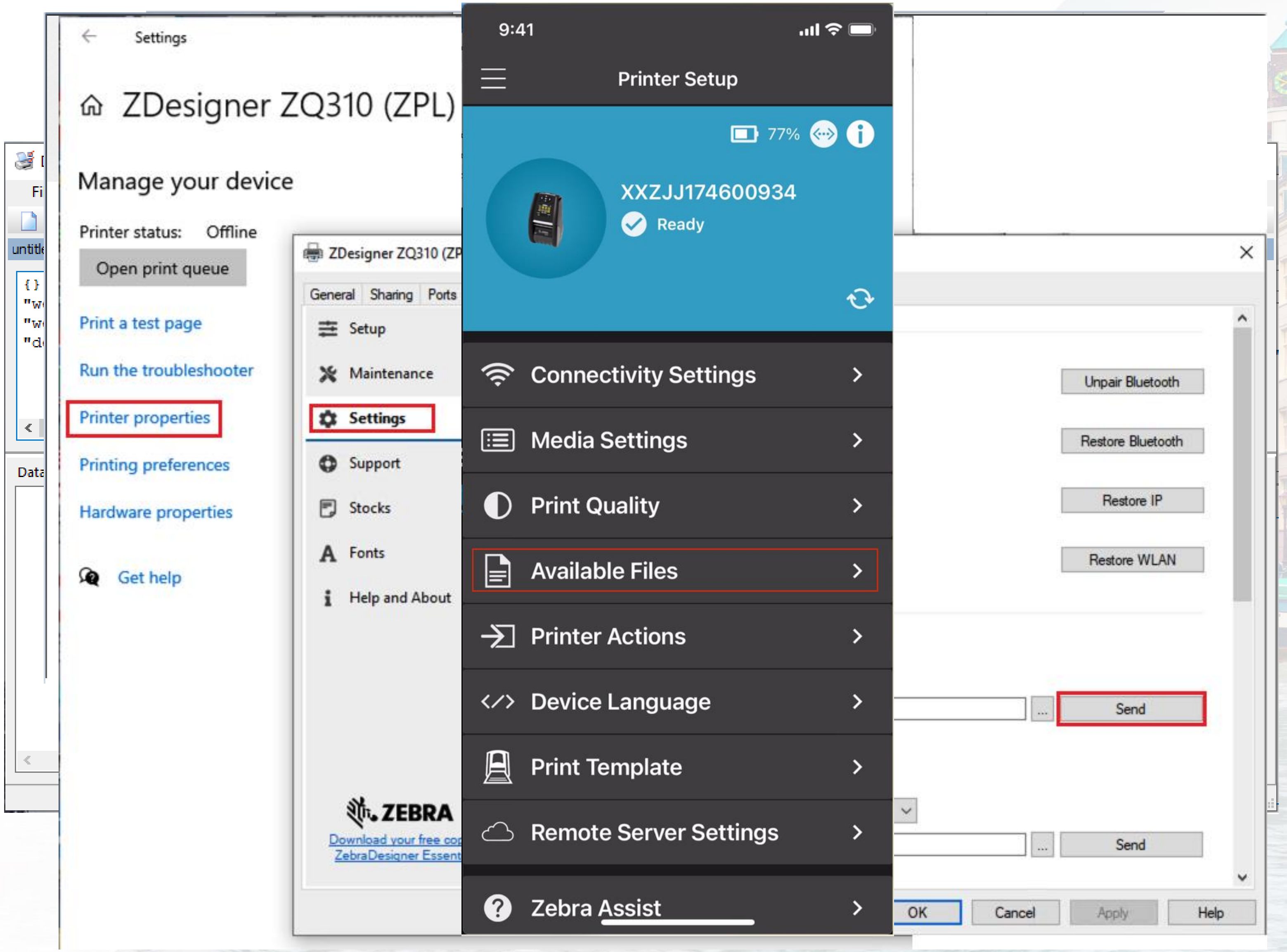


Type	Device ID		Remove
printer	40J133900002	zisvteam4@gmail.com	Remove

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



Printer status: Offline

How to Enroll a Printer? – cont.

Verify the Weblink connection

- Retrive 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.

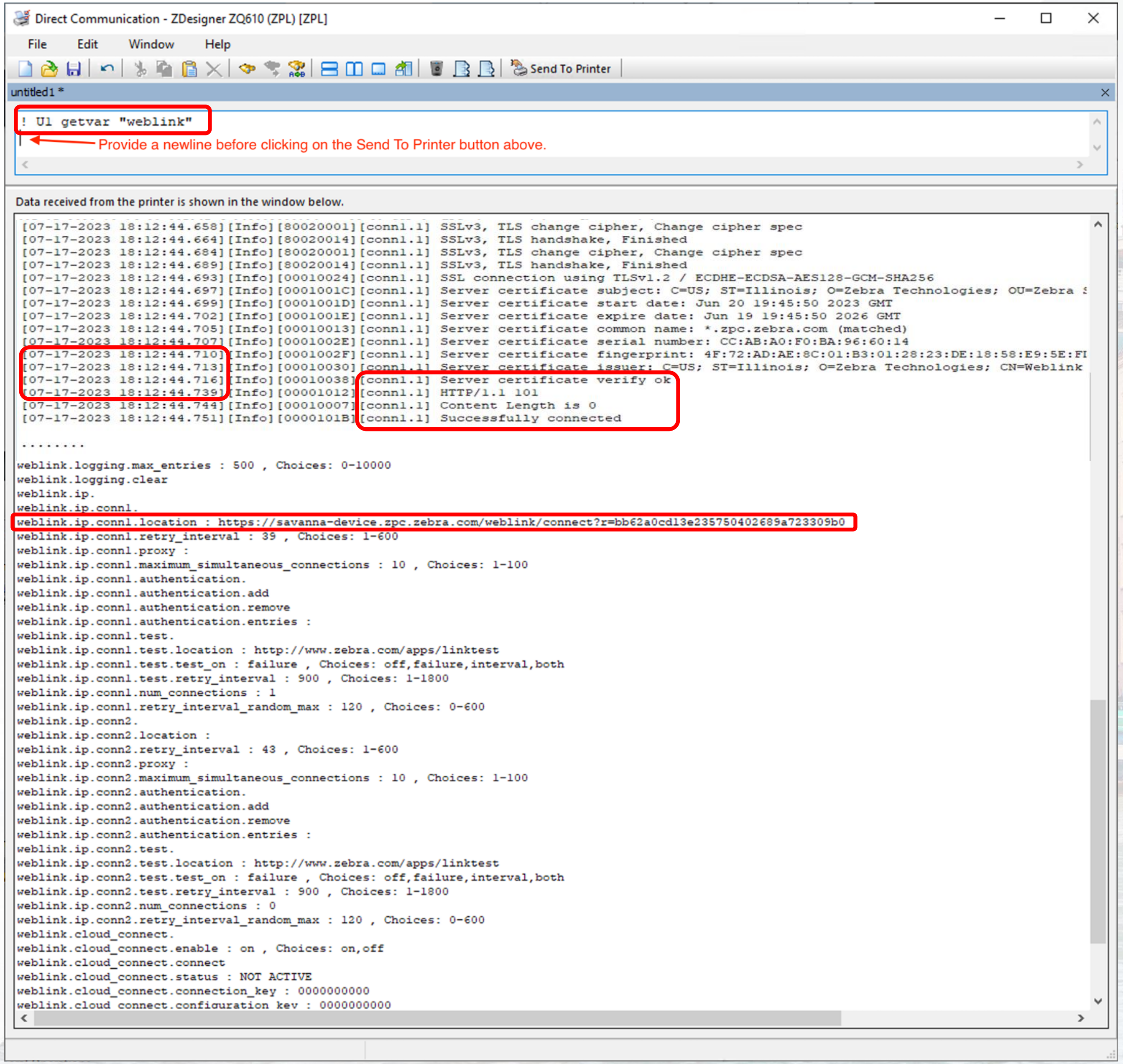
  `[conn1.nn] Successfully connected`

  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?



```
! 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-ECDSA-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 S
[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 : 39 , 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
```

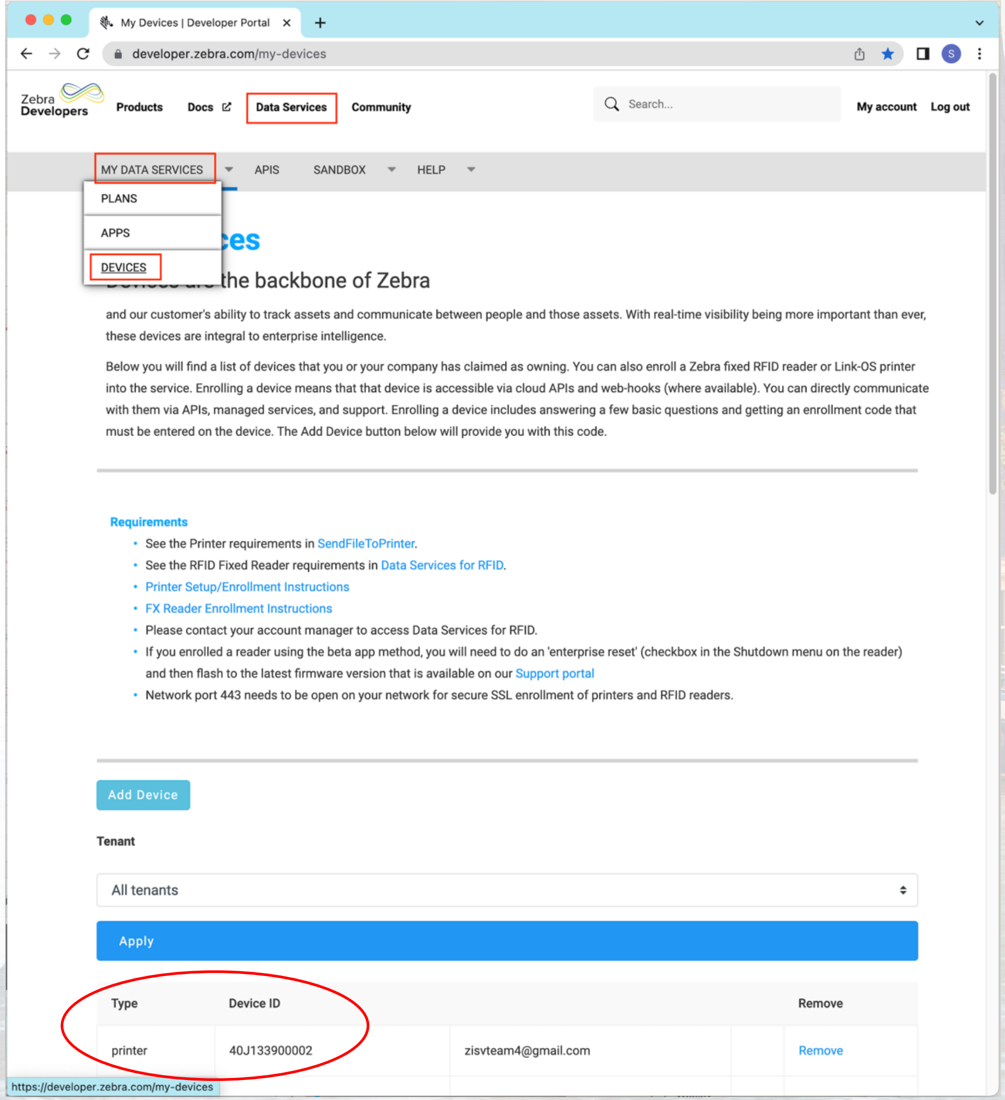
How to Enroll a Printer? – cont.

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



ces

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.

Tenant

All tenants

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

How to Use the SendFileToPrinter API?

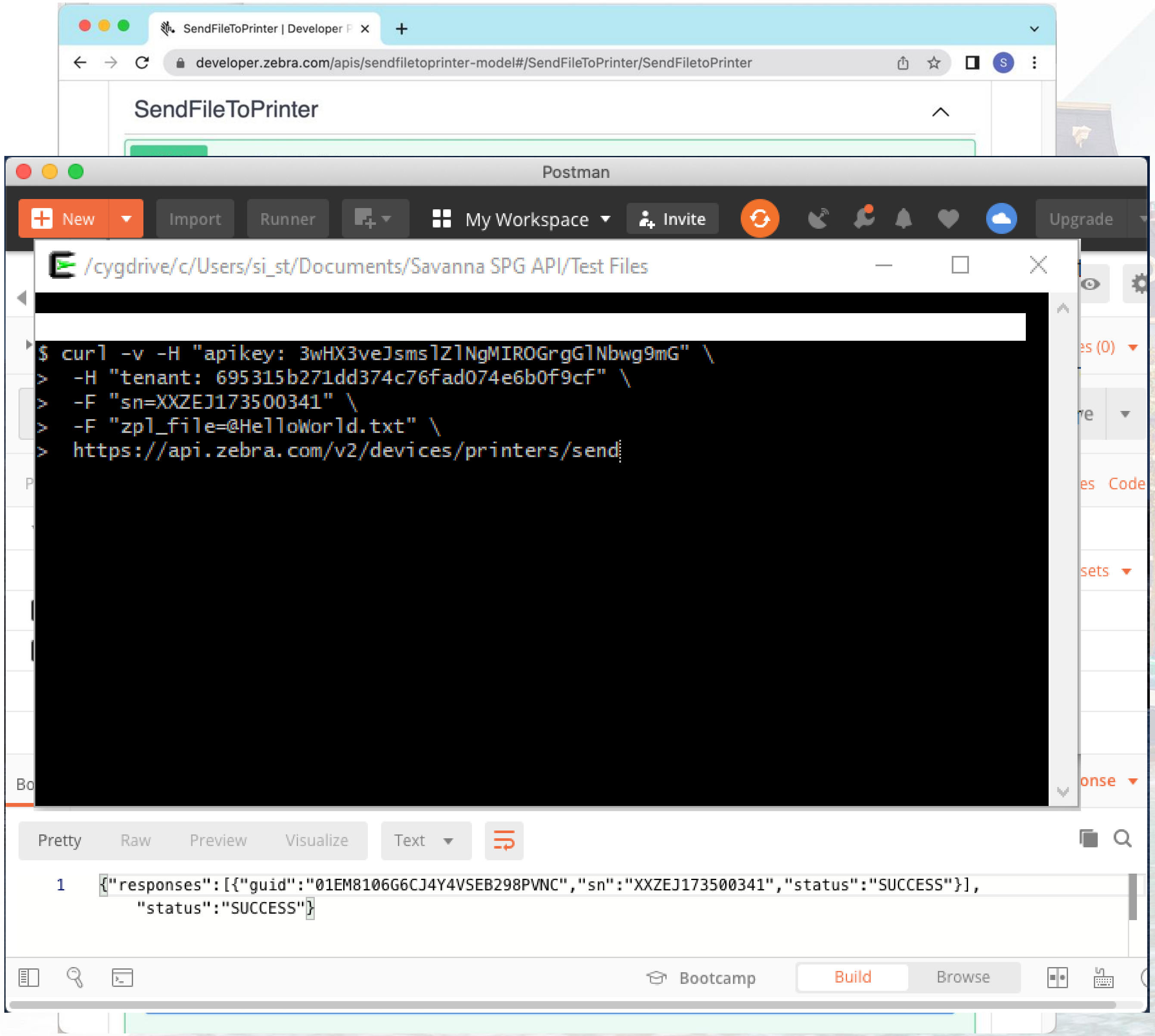
For testing

- [SendFileToPrinter API Portal](https://developer.zebra.com/apis/sendfiletoprinter)

Direct Link: https://developer.zebra.com/apis/sendfiletoprinter

- Postman

- cURL



developer.zebra.com/apis/sendfiletoprinter-model#/SendFileToPrinter/SendFiletoPrinter

/cygdrive/c/Users/si_st/Documents/Savanna SPG API/Test Files

```
$ curl -v -H "apikey: 3wHX3veJsmslZlNgMIROGrgGlNbwg9mG" \
>  -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 uri = URI.create("https://api.zebra.com/v2/devices/printers/send");

String sn = "XXZJJ174600974"; // Printer serial number example
String apikey = "EvqdePLp2WMpK8ZTqrAGDAGqCzs83Ajn"; // API Key example
String tenant = "695315b272dd374c76fad074e6b0f9cf"; // Tenant ID example

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

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

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

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

   fd = new FormData();
   fd.append("zpl_file", fileToSend); // 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)

- Getting Started with Zebra Data Services (https://developer.zebra.com/docs/getting-started)

- Printer Setup Guide (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)

DevCon 2025

Connect | Learn | Build

Printer Profile Manager Enterprise (PPME) Cloud Edition

Part of Zebra Print DNA

Zebra
Print DNA™

Your Printer’s Build-In Advantage

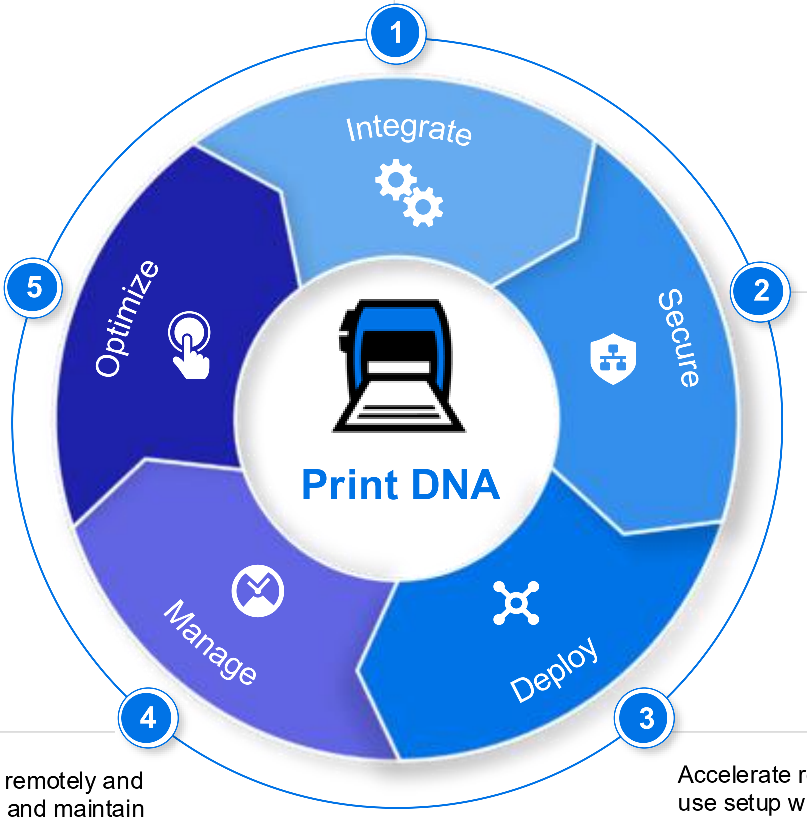
Zebra
Print DNA™

Your Printer's Built In Advantage

Print DNA is a suite of built-in software applications, utilities and tools that deliver unmatched experiences for every kind of user and provide the ability to optimize printers' performance over their entire lifecycle, even as business needs evolve.

www.zebra.com/printdna

DevCon 2025
Connect | Learn | Build



Effortlessly customize and integrate print apps for multiple environments and infrastructures

Give IT the ability to uncover potential printer security vulnerabilities and defend against cyber attacks

Accelerate rollouts with an easy-to-use setup wizard, remote access and streamlined steps

Enable IT to remotely and conveniently manage and maintain printers anywhere, anytime

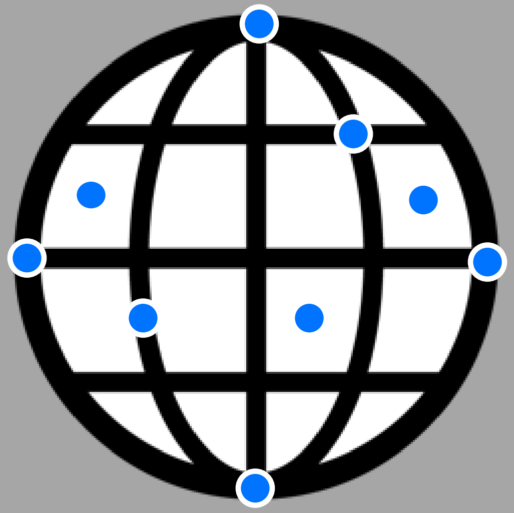
Offer everyone super simple, reliable printing that makes their job much easier to do

Get It Done the Easy Way

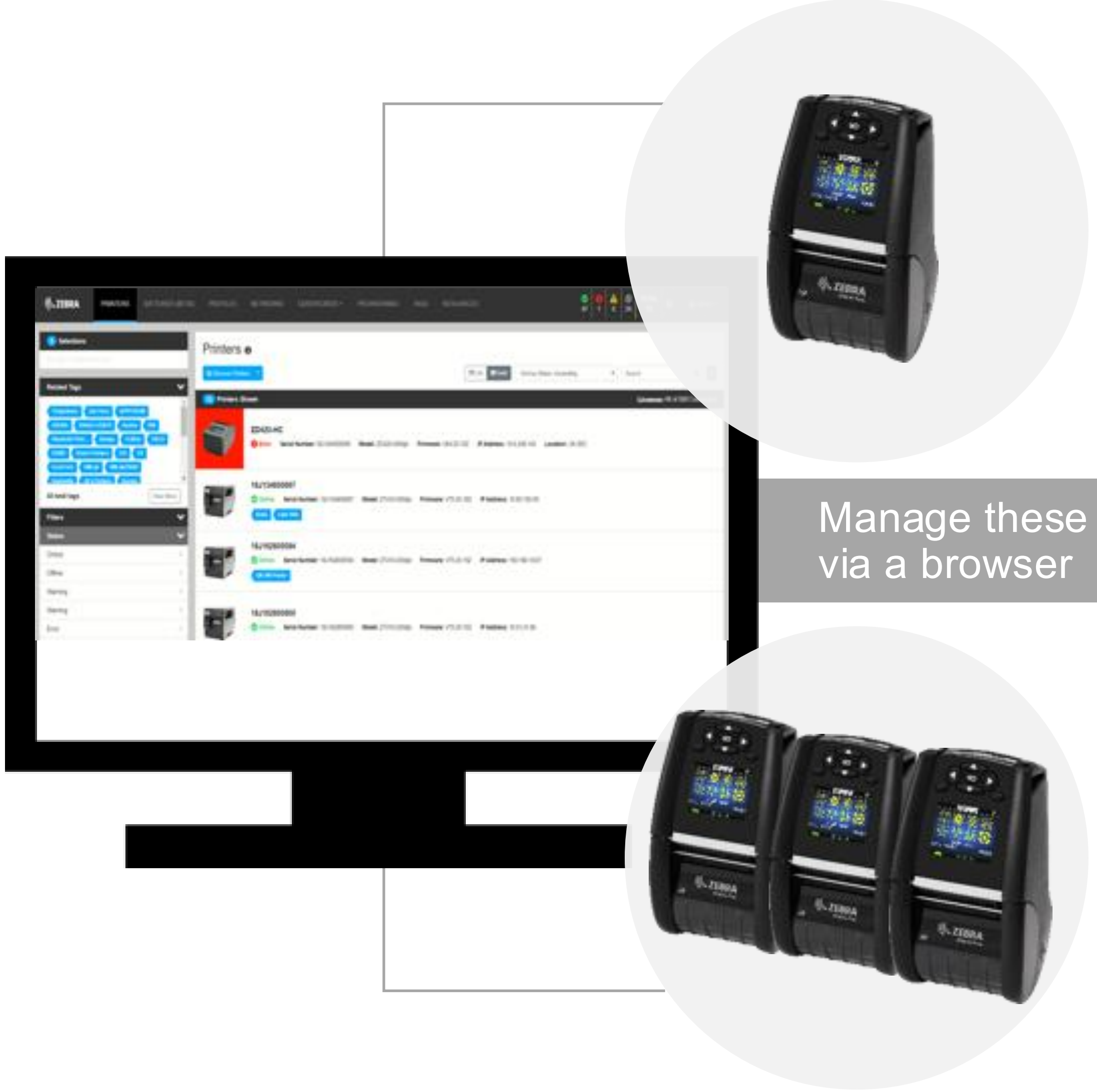
DevCon 2025
Connect | Learn | Build

- **Zebra hosted solution**
- **Exceeds the On-Premise 10K printer limit**
- **Simple Printer Setup**
- **Built-In Security**
- **Printer Profiles**
- **Automation Tools**
- **Real-Time Status Alerts**

Printer Profile Manager Enterprise in Action



Access your
Print DNA printers
anywhere on your global network



Manage individually

Manage these printers via a browser

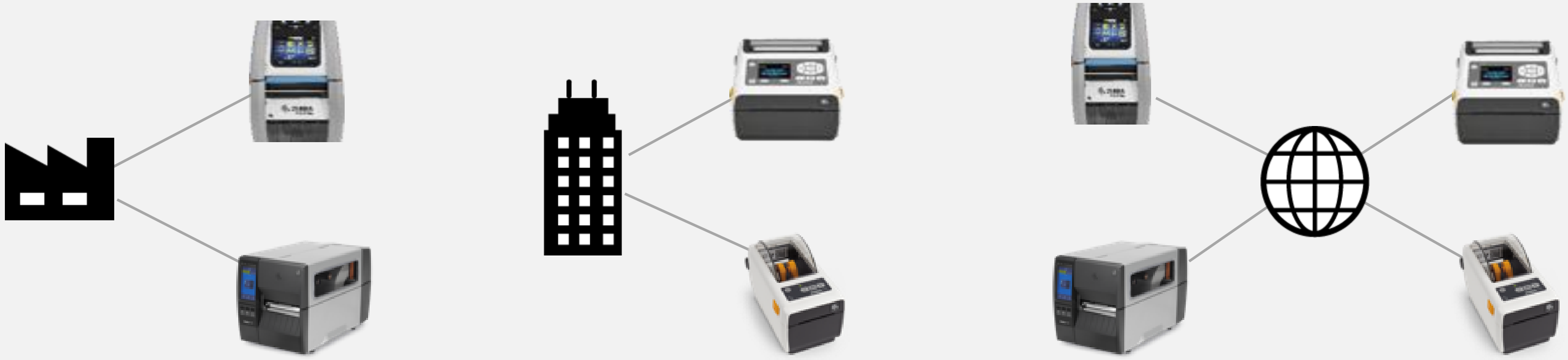
Manage in groups

Easy Printer Grouping

DevCon 2025
Connect | Learn | Build

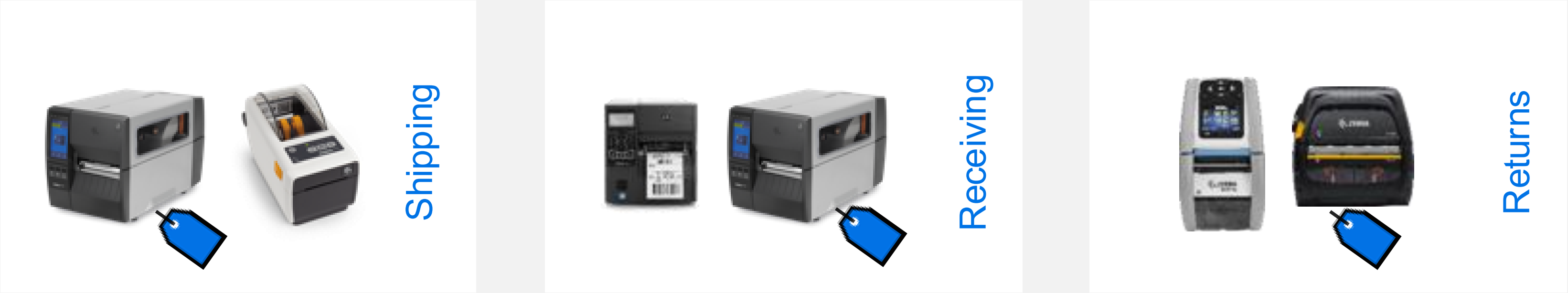
Printer Profile Manager Enterprise Is Your “Eye in The Sky”

Printers can be connected and managed from wherever they are around the globe



“Tags” Let You Organize Your Printers Into Groups

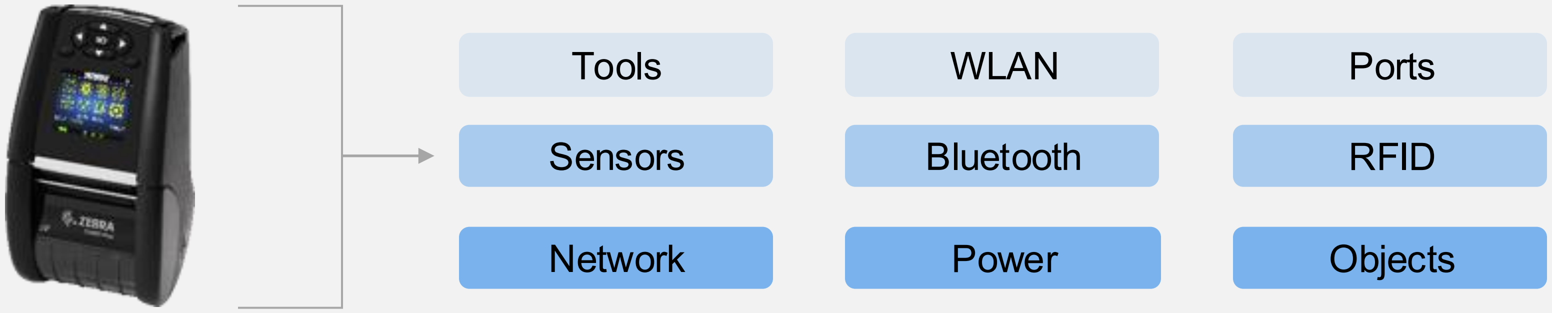
Group/filter printers by “Tags” to make it simple to see just the units you need



Simple Printer Setup

Profile Manager Enterprise Makes Setup “Click Simple”

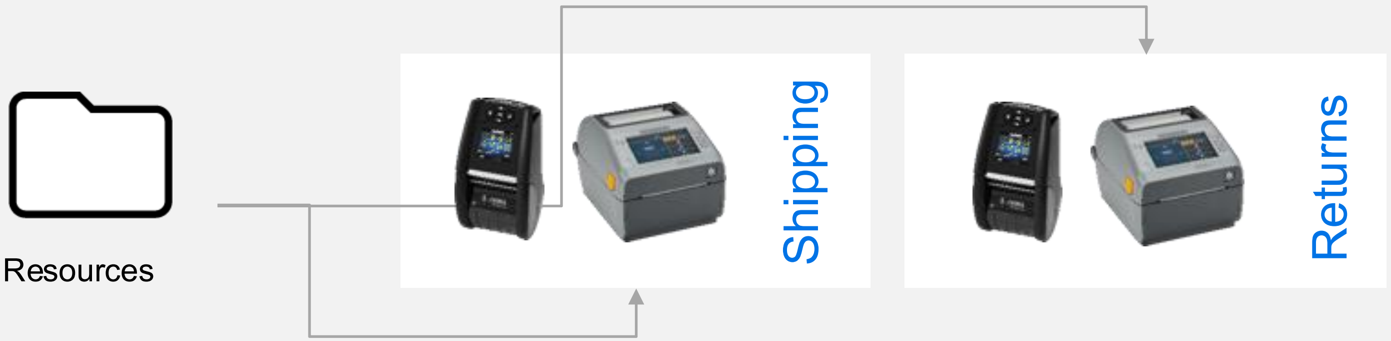
Quickly find what you need with 300+ settings, organized by category



Tools	WLAN	Ports
Sensors	Bluetooth	RFID
Network	Power	Objects

Use Resources Folder to Store Objects

Distribute content to one or many printers by Tag, simplifying multi-printer control

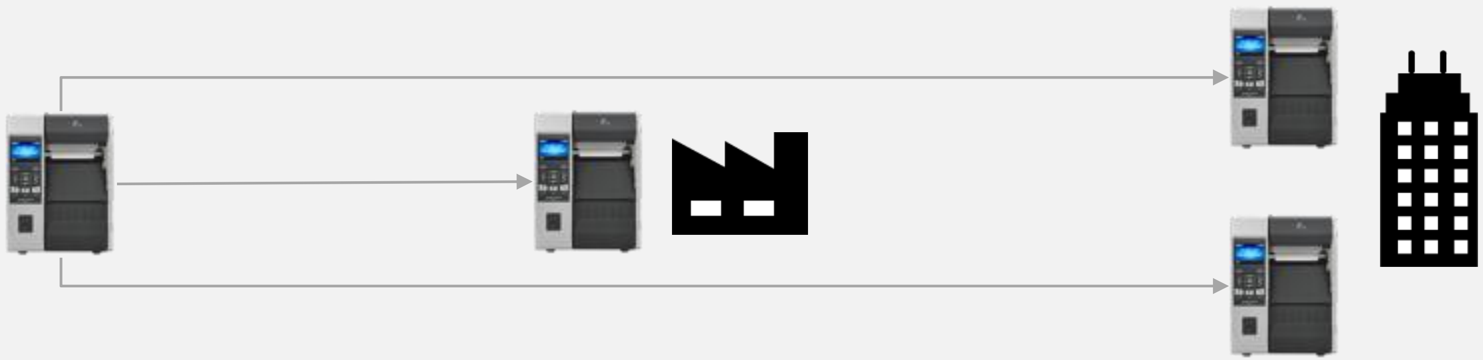


Printer Profiles

DevCon 2025
Connect | Learn | Build

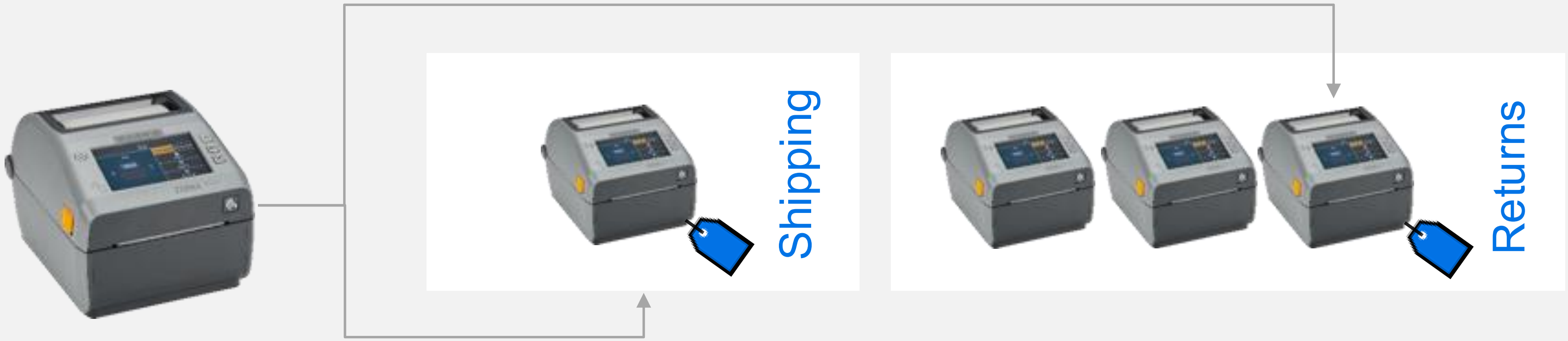
Store Multiple Printer Profiles

Printer profiles contain all the settings and objects that define a printer's configuration



Send Profiles to Printer Groups Using Tags

Distribute content to one or many printers by Tag, simplifying managing many printers

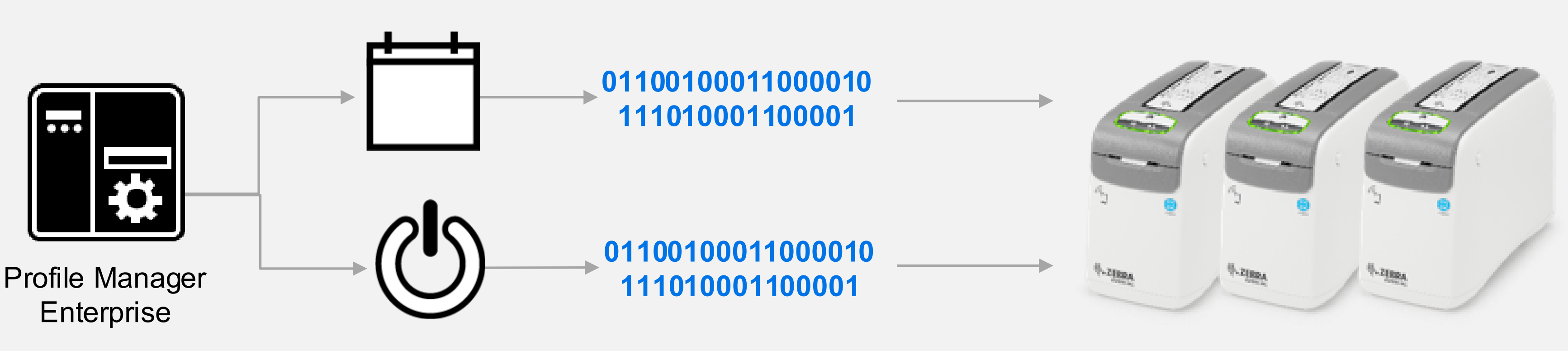


Automation

DevCon 2025
Connect | Learn | Build

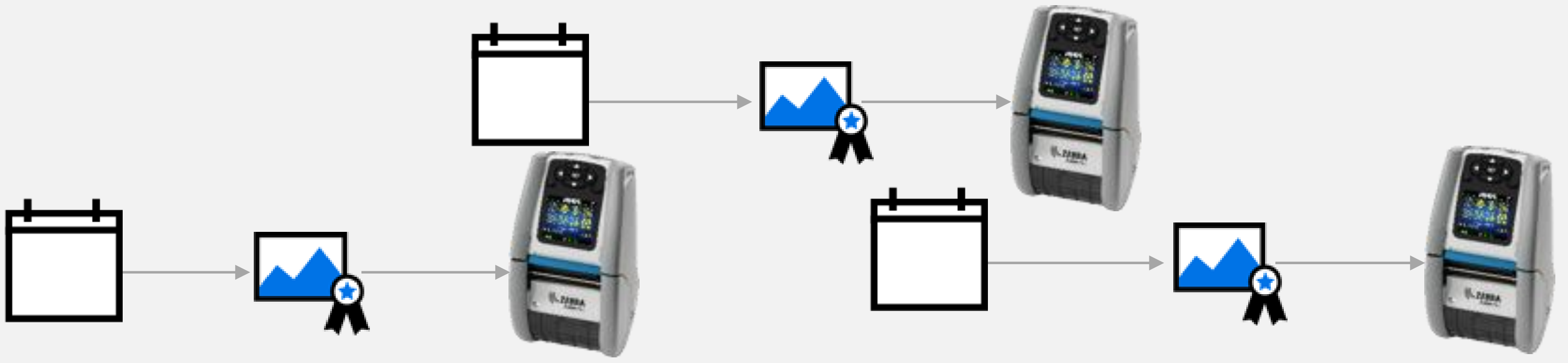
Schedule Printer Updates

Schedule updates by time, date or printer power-up with the “Provisioning” feature



Automate Printer Updates

Automate sending Profiles or Resources outside of worker shifts.

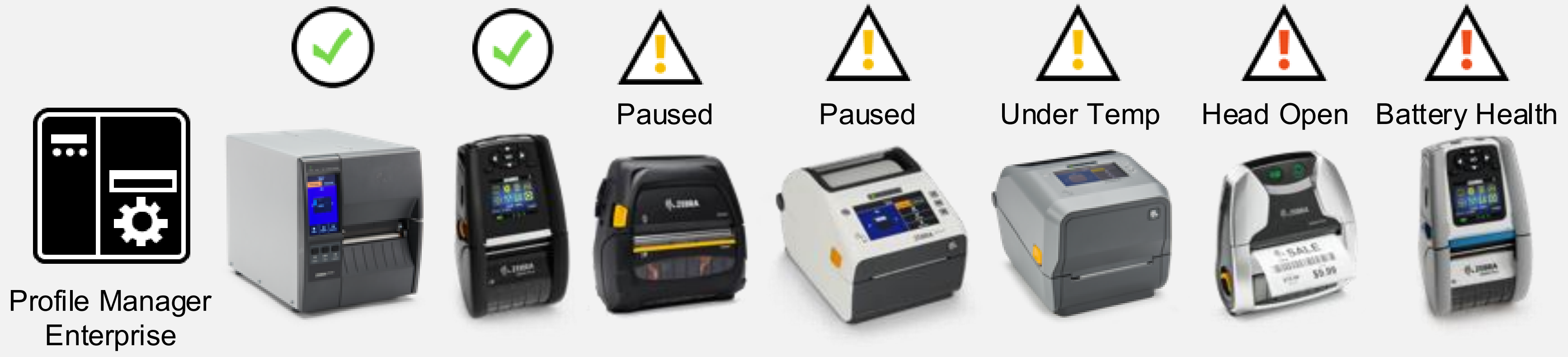


Automation

DevCon 2025
Connect | Learn | Build

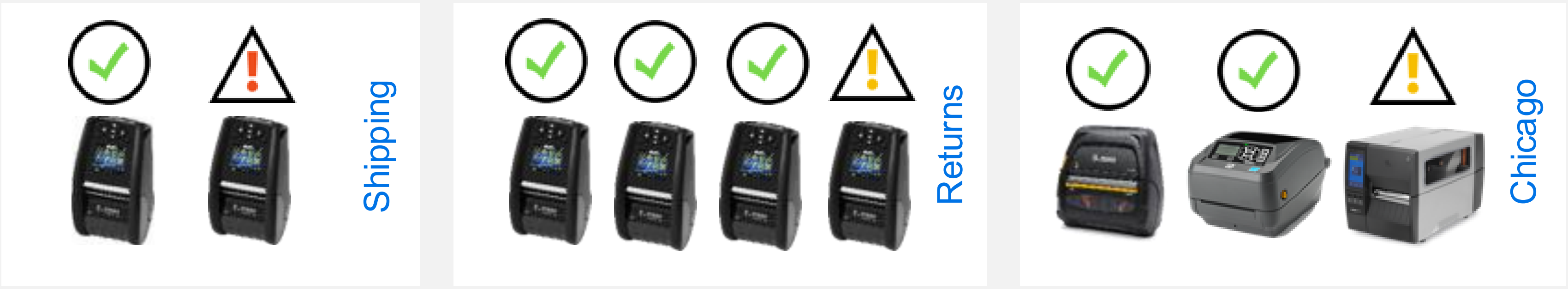
✓ ⚠ ⚠ Real-Time Alerts

Quickly triage issues and get printers back to work with real-time alerts



✓ ⚠ ⚠ Filter Views Based on Status

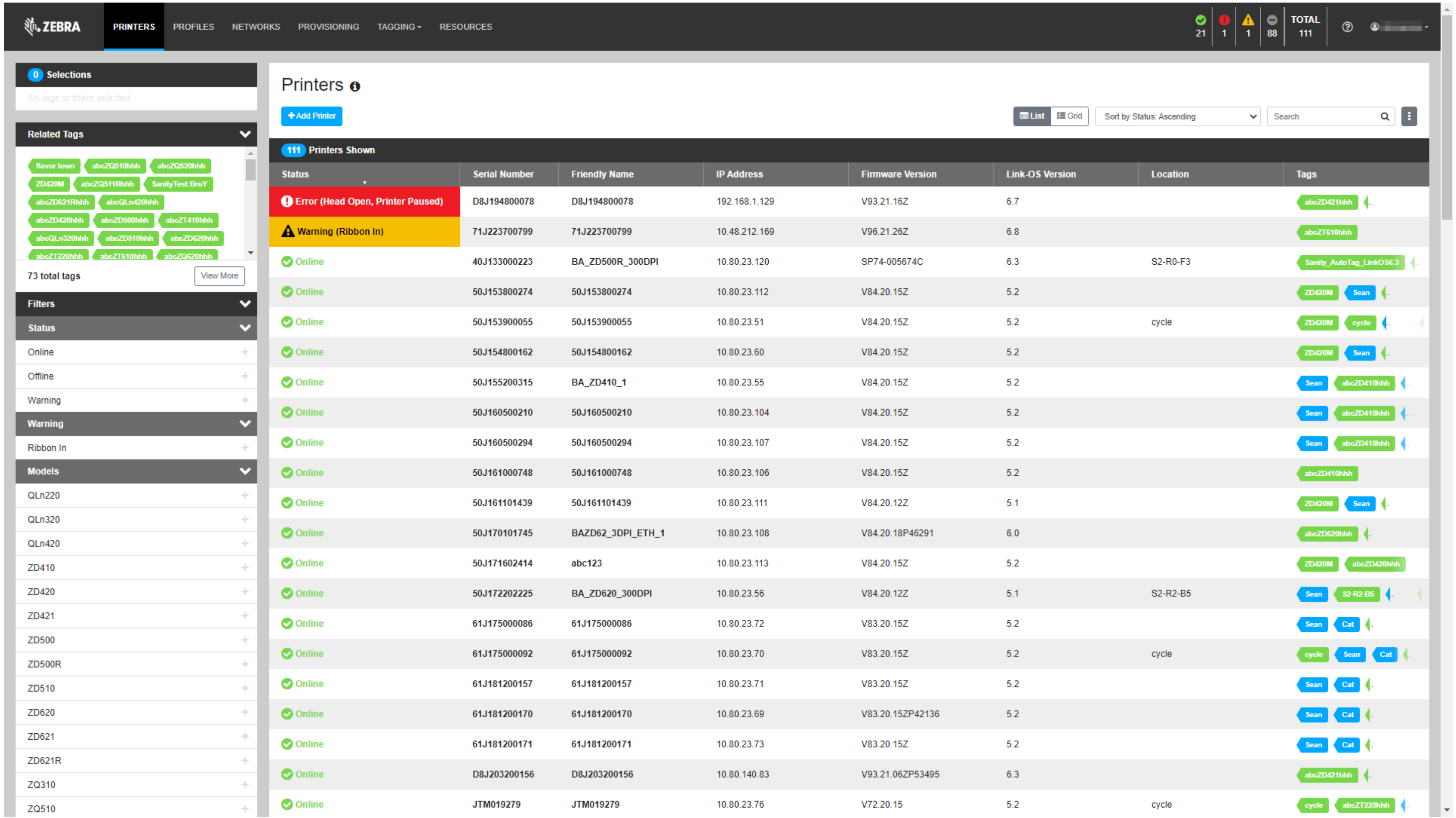
Use multiple filters including Status and Tags to simplify managing large printer fleets



Management Console

DevCon 2025
Connect | Learn | Build

Manage all your printers from a user-friendly dashboard



[redacted]

0 Selections

No tags or filters selected

Related Tags

flavor town | abcZQ510hhh | abcZQ520hhh | ZD420M | abcZQ511Rhhh | SanityTest:BruY | abcZD621Rhhh | abcQLn420hhh | abcZD420hhh | abcZD500hhh | abcZT410hhh | abcQLn320hhh | abcZD510hhh | abcZD620hhh | abcZT220hhh | abcZT610hhh | abcZQ620hhh

73 total tags   View More

Filters

Status
- Online +
- Offline +
- Warning +

Warning
- Ribbon In +

Models
- QLn220 +
- QLn320 +
- QLn420 +
- ZD410 +
- ZD420 +
- ZD421 +
- ZD500 +
- ZD500R +
- ZD510 +
- ZD620 +
- ZD621 +
- ZD621R +
- ZQ310 +
- ZQ510 +

Printers

+ Add Printer

List | Grid   Sort by Status: Ascending   Search

111 Printers Shown

Status	Serial Number	Friendly Name	IP Address	Firmware Version	Link-OS Version	Location	Tags
Error (Head Open, Printer Paused)	D8J194800078	D8J194800078	192.168.1.129	V93.21.16Z	6.7		abcZD421hhh
Warning (Ribbon In)	71J223700799	71J223700799	10.48.212.169	V96.21.26Z	6.8		abcZT610hhh
Online	40J133000223	BA_ZD500R_300DPI	10.80.23.120	SP74-005674C	6.3	S2-R0-F3	Sanity_AutoTag_LinkOS6.3
Online	50J153800274	50J153800274	10.80.23.112	V84.20.15Z	5.2		ZD420M Sean
Online	50J153900055	50J153900055	10.80.23.51	V84.20.15Z	5.2	cycle	ZD420M cycle
Online	50J154800162	50J154800162	10.80.23.60	V84.20.15Z	5.2		ZD420M Sean
Online	50J155200315	BA_ZD410_1	10.80.23.55	V84.20.15Z	5.2		Sean abcZD410hhh
Online	50J160500210	50J160500210	10.80.23.104	V84.20.15Z	5.2		Sean abcZD410hhh
Online	50J160500294	50J160500294	10.80.23.107	V84.20.15Z	5.2		Sean abcZD410hhh
Online	50J161000748	50J161000748	10.80.23.106	V84.20.15Z	5.2		abcZD410hhh
Online	50J161101439	50J161101439	10.80.23.111	V84.20.12Z	5.1		ZD420M Sean
Online	50J170101745	BAZD62_3DPI_ETH_1	10.80.23.108	V84.20.18P46291	6.0		abcZD620hhh
Online	50J171602414	abc123	10.80.23.113	V84.20.15Z	5.2		ZD420M abcZD420hhh
Online	50J172202225	BA_ZD620_300DPI	10.80.23.56	V84.20.12Z	5.1	S2-R2-B5	Sean S2-R2-B5
Online	61J175000086	61J175000086	10.80.23.72	V83.20.15Z	5.2		Sean Cat
Online	61J175000092	61J175000092	10.80.23.70	V83.20.15Z	5.2	cycle	cycle Sean Cat
Online	61J181200157	61J181200157	10.80.23.71	V83.20.15Z	5.2		Sean Cat
Online	61J181200170	61J181200170	10.80.23.69	V83.20.15ZP42136	5.2		Sean Cat
Online	61J181200171	61J181200171	10.80.23.73	V83.20.15Z	5.2		Sean Cat
Online	D8J203200156	D8J203200156	10.80.140.83	V93.21.06ZP53495	6.3		abcZD421hhh
Online	JTM019279	JTM019279	10.80.23.76	V72.20.15	5.2	cycle	cycle abcZT220hhh

Management Console

Purchase a subscription based on how many printers you have now, and add more as your business grows

Part Number	Description	Terms
SW-PMCLOUD-TRIAL	60-Day Trial of Zebra Printer Management Cloud	60-day trial
SW-PMCLOUD-1YR	1-Year Subscription for 1 Printer for Zebra Printer Management Cloud	1-year subscription
SW-PMCLOUD-1YR-R	1-Year Renewal Subscription for 1 Printer for Zebra Printer Management Cloud	1-year subscription renewal

The SW-PMCLOUD-1YR part number should be ordered in quantities according to the tier quantities below.

Tier Quantities:
100, 250, 500, 1000, 1500, 2500, 3500, 5000, 6500, 8000.
For quantities beyond 8000, or a deviation from this tier structure, contact Zebra ATS Product Management.

DevCon 2025

Connect | Learn | Build

Questions?

DevCon 2025

Connect | Learn | Build

Thank You