VMware AirWatch Peripheral API Guide

Configuring and Using Peripheral APIs with AirWatch

AirWatch v9.1

Have documentation feedback? Submit a Documentation Feedback support ticket using the Support Wizard on support.air-watch.com.

Revision Table

The following table displays revisions to this guide since the release of AirWatch v9.1.

Date	Reason
April 2017	Initial upload. Document posted for AirWatch v9.1 GA.

Table of Contents

Chapter 1: Overview

Introduction to Peripheral API Guide

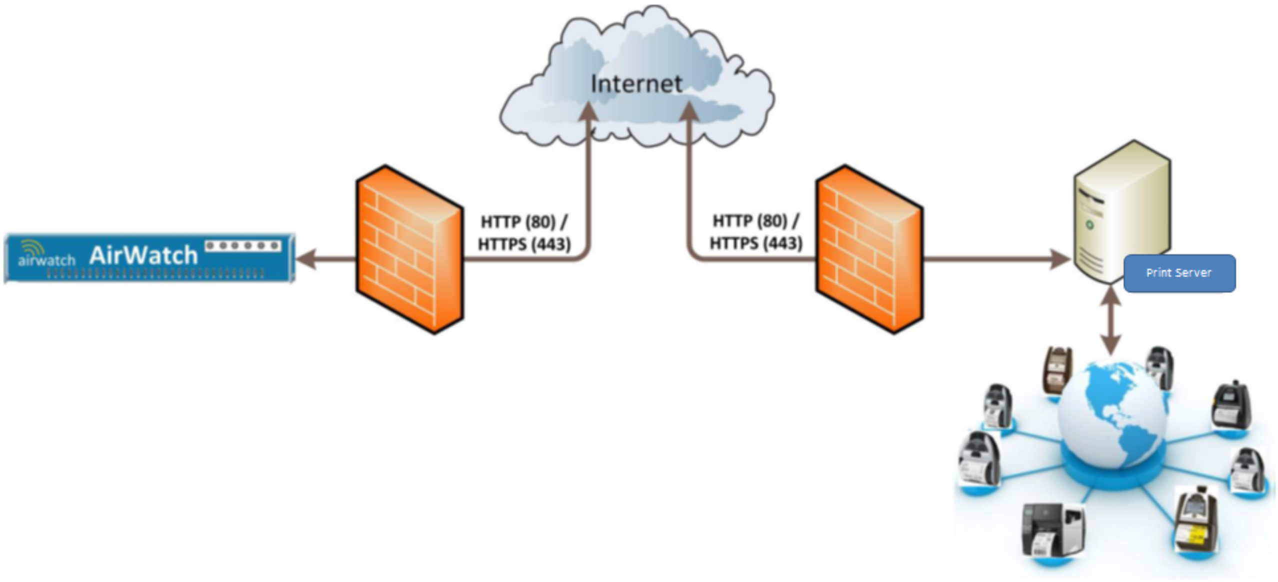
AirWatch created a menu and dashboard in the AirWatch Console to help manage your external peripheral devices. This guide focuses on peripheral-specific information and actions such as monitoring printers, gathering reports, monitoring alerts, and pushing profiles and files, including firmware upgrades. It provides you with an easy method for integrating your peripheral devices with AirWatch and the ability to manage them securely through the AirWatch Console.

To integrate peripherals, AirWatch provides a collection of APIs which allow external peripheral devices to communicate information to and receive commands from the AirWatch Console. Available peripheral APIs currently include initialization, enrollment and activation, as well as issuing and receiving commands.

This guide is intended for application developers and provides an understanding of the design and architecture of the peripheral API library to facilitate a custom development and integration with AirWatch.

Use Case: Printer Management

AirWatch does not communicate directly with printers. Instead, AirWatch communicates to the Print Server, which in-turn communicates all requests to the printers through Wi-Fi or Ethernet. For a diagram that illustrates this, refer to the *AirWatch Integrated with Print Server and Printers* diagram below.

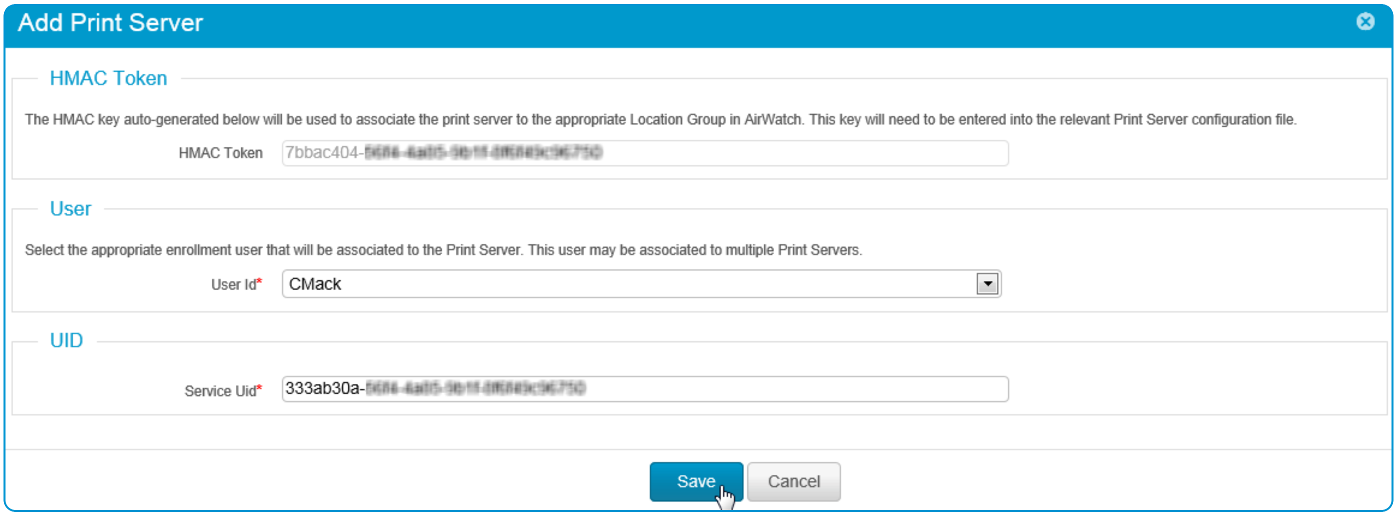


The illustration above shows only one of many different possible configurations. A Print Server and AirWatch server should be installed on different hardware or any other server that allows communication between both servers. It is not recommended to install the Print Server and AirWatch on the same hardware. This illustration shows the Print Server and AirWatch server located in different locations or networks so they require HTTPS to communicate through the firewall. If both severs were located within the same firewall, then the servers can use HTTP.

Integrating a Print Server with AirWatch

To begin the process of managing printers, you must first establish a communication link between AirWatch and the Print Server. The following identifiers are used to initiate integration between AirWatch and the Print Server:

- **HMAC Token** – Auto-generates a key/token that is used to encrypt informiton exchanged between Print Server and AirWatch, as well as associate the Print Server to the appropriate Organization Group in the AirWatch server.
  - AirWatch automatically generates the HMAC token. You will need to copy the HMAC Token and paste it into the Print Server installation screen
- **User** –The user to which the Printer is associated to the Print Server.
  - This is the **User ID** for an enrollment user that is associated with the Print Server. You can associate more than one Print Server with the same User Id.
- **UID** – Unique Print Server Identifier.
  - The UID field is empty because it is not generated by AirWatch –it is generated by the Print Server. The UID needed for this field automatically generates when you install/configure the Print Server. You will need to copy the UID from the Print Server installation screen and paste it into the AirWatch UID field.
  - Once you upload the HMAC Token to the Print Server and upload the UID to the AirWatch server, both servers have the items needed to establish communication. Of course, each Print Server might have other requirements, such as passing a certificate to authenticate so refer to the specific Print Server instructions for additional details.



Add Print Server

HMAC Token

The HMAC key auto-generated below will be used to associate the print server to the appropriate Location Group in AirWatch. This key will need to be entered into the relevant Print Server configuration file.

HMAC Token: 7bbac404-[redacted]

User

Select the appropriate enrollment user that will be associated to the Print Server. This user may be associated to multiple Print Servers.

User Id*: CMack

UID

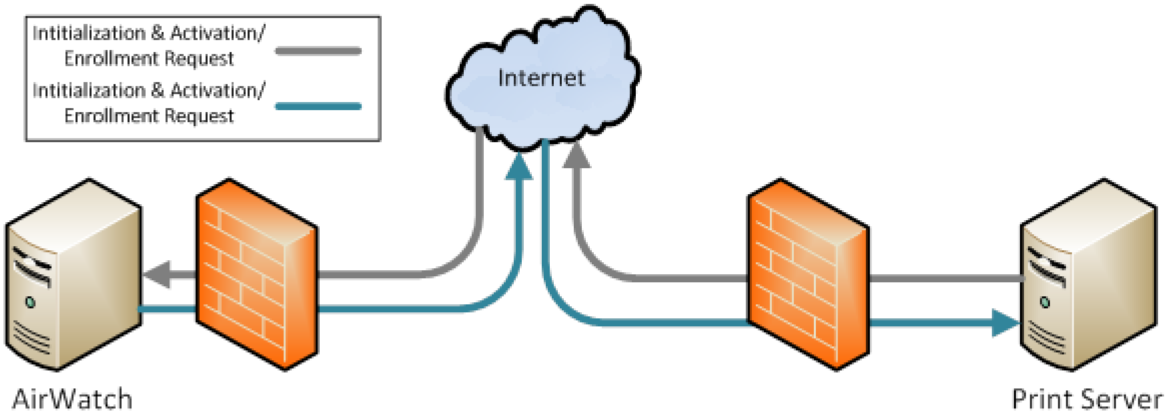
Service Uid*: 333ab30a-[redacted]

Printer Initialization, Activation, and Enrollment

Once the Print Server and AirWatch are communicating, printers are successfully activated and enrolled within the proper AirWatch Organization Group via a request that contains the following key information:

- Peripheral Type
- Device Unique ID (UID)
- Device Name
- Model
- Serial Number
- Group ID (corresponds to the Organization Group the print server was configured for)

Similar to the initialization request, AirWatch returns a success/failure message to the Print Server as confirmation that all information was received.

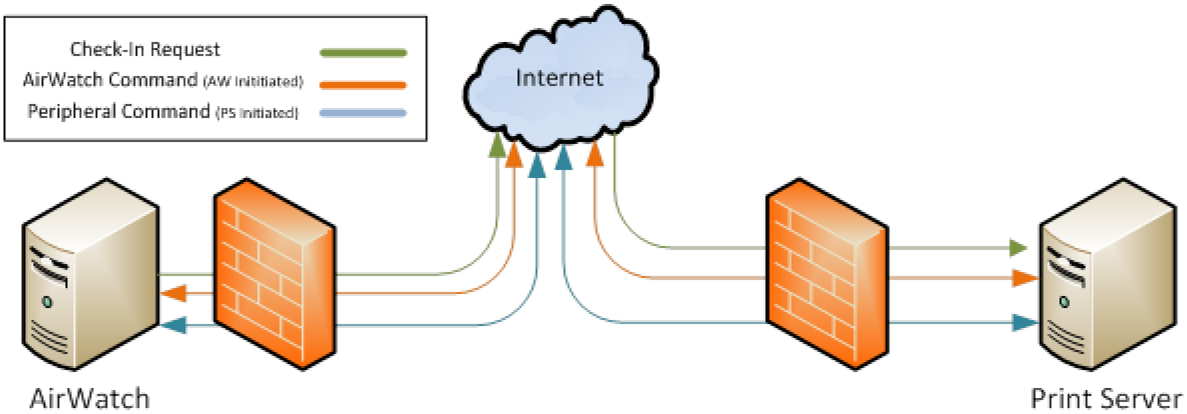


AirWatch and Peripheral Commands

Once Activated/Enrolled, the printer has the ability to pick-up AirWatch Commands (i.e., profiles or files) as commands are published, as well as use Peripheral Commands to report Alerts (e.g., paper out, ribbon out, etc.).

When a profile or file is published, AirWatch first sends out a blank check-in request to the notification URL (**https://api.acme.com/acmeprinters/wakeup/<deviceUDID>**), which triggers the Print Server to contact the AirWatch server for commands. AirWatch in-turn responds with the profile/file information.

The Print Server initiates alerts messages. The Print Server does not require a check-in request to begin communication with AirWatch. Alerts use the Peripheral Command API, and include device identifying information, along with the appropriate alert. As with Initialization and Activation APIs, the Peripheral Command API sends a success/failure response to the Print Server upon receipt of the alert.


Check-In Request
AirWatch Command (AW Intitiated)
Peripheral Command (PS Initiated)
Internet
AirWatch
Print Server

Initialization

The following API is used to initiate a “handshake” with the AirWatch web service when the peripheral device first comes online. This handshake provides AirWatch with relevant information about the peripheral such as the server (unique) ID, server version and web service URL.

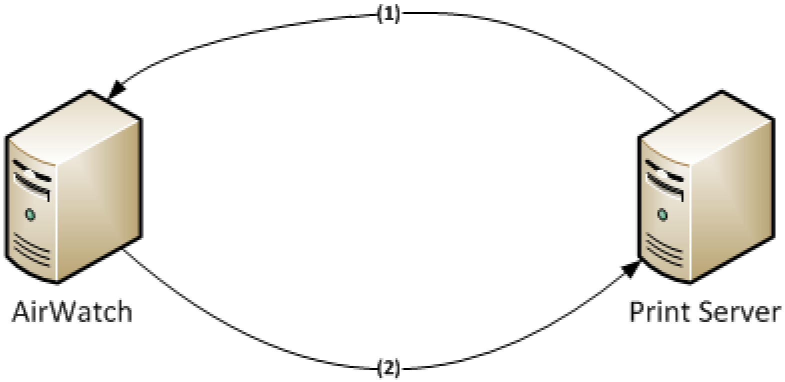
Process Flow:

1. Request sent to **https://<AW Device Services URL/deviceservices/peripheralservice/v1/register**

- Contains {Print Server UID, Notification Endpoint, GroupID}

2. Response sent to Print Server endpoint (this endpoint is determined by peripheral vendor)

- Contains {Enrollment URL, Command URL, Alert URL}



Resource URL: https://api.acme.com/deviceservices/peripheralservice/v1/register/

Request:

```
{
"uid": "Unique peripheral service id",
"NotificationEndpoint": "https://printservice/notify/{authToken}",
"ServerVersion": "0.1",
"GroupId":"Acme"
}
```

Response (Success):

```
{
"ServerVersion":"6.1",
"EnrollmentURL":"https://api.acme.com/deviceservices/peripheralservice/enrol
ldevice",
"CommandURL":"
https://api.acme.com/deviceservices/peripheralservice/checkin",
"AlertURL":" https://api.acme.com/deviceservices/peripheralservice/alert",
"Status":"0"
}
```

Response (Failure):

```
{
"Status": "1",
"ErrorCode": ""
}
```

Note: Generate and obtain the print service’s HMAC key needed to make this request AirWatch HMAC signed from the AirWatch Console.

Enrollment and Activation

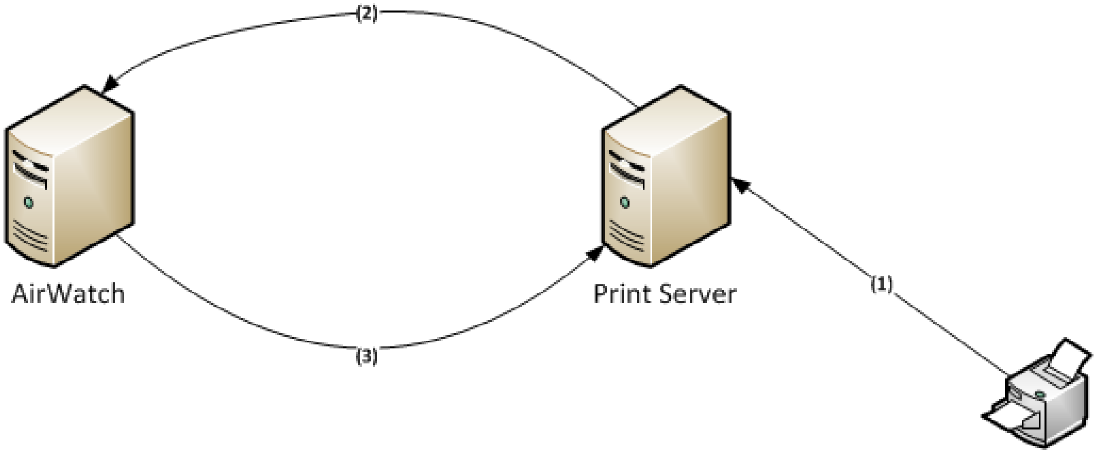
The following API is used to enroll the peripheral and any additional devices the peripheral manages into AirWatch.

Process Flow:

1. Printer is manually configured to communicate with Print Server
2. Request sent to: **https://<AWDeviceServicesURL/deviceservices/peripheralservice/enrolldevice**

- Contains {Print Server UID, Device UDID, Printer Name, Printer Model, Serial Number, GroupID}

3. Response contains success or failure.



Resource URL: https://api.acme.com /deviceservices/peripheralservice/enrolldevice/

Request:

```
{
"ServiceUid": "Unique peripheral service id",
"Type":"Printer",
"deviceUid": "Printer uid",
"Name":"Tradefloor basement printer1",
"Model":"ACME",
"SerialNumber": "{serialnumber}",
"GroupId":"Acme"
}
```

Response (Success):

```
{
"Status": "0"
}
```

Response (Failure):

```
{
"Status": "1",
"ErrorCode": ""
}
```

Note: Generate and obtain the print service’s HMAC key needed to make this request AirWatch HMAC signed from the AirWatch Console.

AirWatch Commands

The following API is used to push commands to the peripheral server. For example, AirWatch can push and install a profile to the device which includes peripheral settings.

Process Flow:

1. Wakeup command sent to notification URL (**https://api.acme.com/acmeprinters/wakeup/<deviceUDID>**).

2. AirWatch Command Request (check-in) received by AirWatch from Print Server.

- Sent to **https://<AW Device Services URL/deviceservices/peripheralservice/checkin**

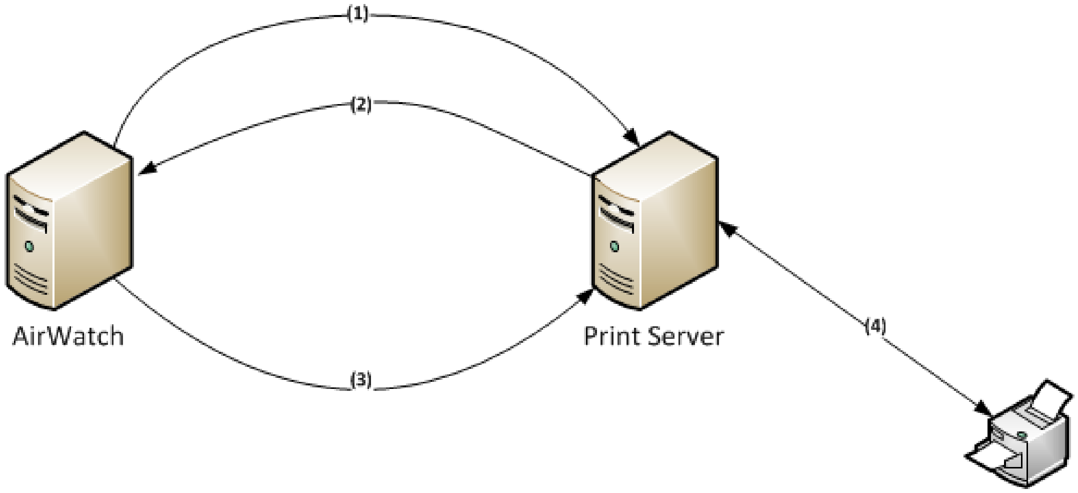
- If this check-in request is the result of a wakeup command, the fields {CommandUID, CommandType, ResponsePayload, ErrorMessage} will be null. {Status} will be "Idle".

- If this request is an acknowledgement of a previous AirWatch Command Response (such as a profile installation), {CommandUID, CommandType} will be populated with the appropriate values for the command for which the Print Server is acknowledging. {Status} will be acknowledged. If AirWatch has additional commands queued for the device, we will respond with a new {CommandUID} for the new command (Print Server will consequently respond with an acknowledgment). This process continues until there are no commands in queue, at which time no response is sent back to the Print Server.

3. AirWatch Command Response sent to print server.

- Contains {CommandUID, DeviceUID, CommandType, CommandPayload}

- Print server will respond with an acknowledgment (using request template)



Resource URL: https://api.acme.com /deviceservices/peripheralservice/checkin/

Request:

```
{
"ServiceUid": "Unique peripheral service id",
"DeviceUid": "Printer uid",
"CommandUid": "uid of the command that this response is for (if any)",
“CommandType”:”InstallProfile|PrinterQuery”
"Status": "Idle|Acknowledged|Error|NotNow|CommandFormatError",
"ResponsePayload":
    {
    The Response payload is primarily used for gathering device information
    for a
    request that is in response to a PrinterQuery.
    },
"ErrorMessage":"Error message if any (optional)"
}
```

Response for InstallProfile (Success):

```
{
    "CommandUid":"uid of the command",
    "DeviceUid":"Printer uid",
    "CommandType":"InstallProfile",
    "CommandPayload":
        {
        "ProfileName":"Printer Settings1",
        “ProfileId”: “Unique profile id”,
        "Version":"1",
        “Network”:
        {
            “key1”: ”value1”,
            “key2”: “value2”
        }
        “ZTE”
        {
            “key1”: “value1”,
            “key2”: “value2”
        }
    }
}
```

Response for InstallFile (Success):

```
{
    "CommandUid":"uid of the command",
    "DeviceUid":"Printer uid",
    "CommandType":"InstallFile",
    "CommandPayload":
    {
        "FileName":"Printer File name",
        "FileId":"Unique id",
        "Version":"1.1",
        "CheckSum":"24ef947eaced8cf3d3f93d8e970dfd9c17cfe3ee", --SHA1 hash of
        the template
        binary file
        "DownloadUrl":"https://<deviceServicesURL>/DeviceServices/12112234343
        4545656567
        676767/file1", -- this url is an example, and is valid only for few
        hrs
    }
}
```

Response for Device Reset (Success):

```
{
"CommandUid":"uid of the command",
"DeviceUid":"Printer uid",
"CommandType":"Reset",
}
```

Response for Factory Reset (Success):

```
{
"CommandUid":"uid of the command",
"DeviceUid":"Printer uid",
"CommandType":"ResetDefaults",
}
```

Response (Failure):

A blank response indicates no additional commands for the device.

> **Note**: Generate and obtain the print service's HMAC key needed to make this request AirWatch HMAC signed from the AirWatch Console.

Both the Device and Factory Reset commands do not require acknowledgment.

Possible Status

Idle	The device is idle. There is no status. (Idle would be sent in the first request.)
Acknowledged	The command was received successfully.
Error	An error has occurred. See the ErrorMessage for details.
CommandFormatError	A protocol error has occurred. The command may be malformed.
NotNow	The device received the command, but cannot perform it at this time. The device will poll the server again in the future.

Peripheral Commands

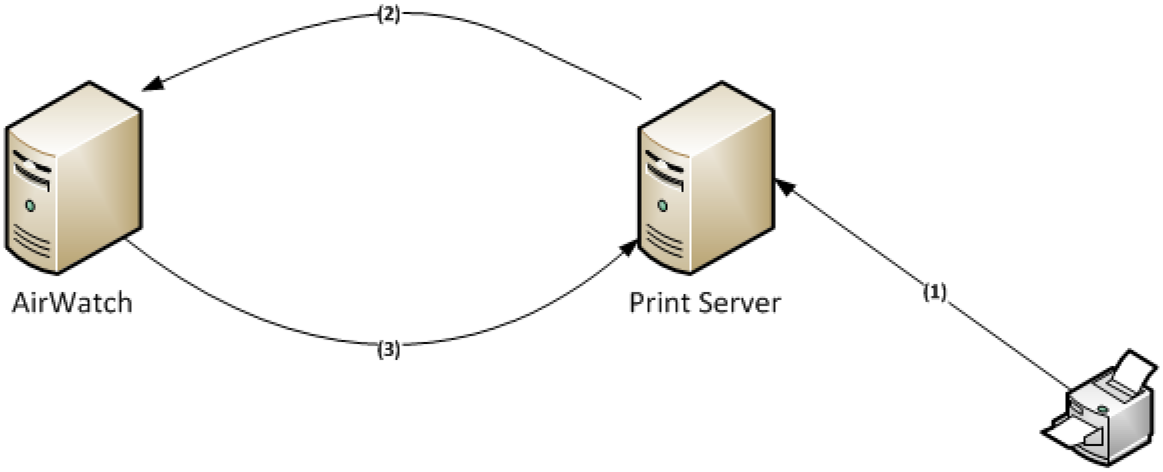
The following API is used to return requested information from the peripheral device to AirWatch including printer information, status and default settings.

Process Flow:

1. Alert information sent from device to print server.
2. Peripheral Command Request sent to **https://<AWDeviceServicesURL/deviceservices/peripheralservice/alert**.

- Contains {ServiceUID, DeviceUDID, AlertType, AlertPayload}

3. Success/Failure response.



Resource URL: https://api.acme.com /deviceservices/peripheralservice/alert/

Request:

```
{
"ServiceUid": "Unique peripheral service id",
"DeviceUid": "Printer uid",
"AlertType": "PrinterStatus",
"AlertPayload":
    {
        "Severity": "1",
        "Message": "Paper is out",
        “Condition”: “Set|Clear”
    }
}
```

Response (Success):

```
{
"Status": "0"
}
```

Response (Failure):

None

Note: Available alert types and severity levels are as follows:

PAPER OUT	SUPPLY TOO HOT	POWER ON	BATTERY LOW
RIBBON OUT	RIBBON IN	CLEAN PRINTHEAD	COLD START
HEAD TOO HOT	REWIND	MEDIA LOW	SGD SET
HEAD COLD	CUTTER JAMMED	RIBBON LOW	
HEAD OPEN	HEAD ELEMENT BAD	REPLACE HEAD	

Generate and obtain the print service's HMAC key needed to make this request AirWatch HMAC signed from the AirWatch Console.