GovPrint interoperability discussion paper

# Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Updated by** | **Comments / main updates** | **Date** |
| 1 | Keith Houghton | Initial version | 15/03/2019 |
| 1.1 | Keith Houghton | Updates to align with Technology Code of Practice | 05/06/2019 |
| 2 | Keith Houghton | Update based on supplier feedback and internal review | 15/10/2019 |

# Contents

[**Version Control**](#_30j0zll) **1**

[**Contents**](#_1fob9te) **1**

[**Introduction**](#_3znysh7) **3**

[**Problem Statement**](#_2et92p0) **5**

[**Interoperability Functions**](#_tyjcwt) **6**

[**Candidate Open Standards**](#_3dy6vkm) **6**

[Internet Printing Protocol](#_1t3h5sf) 6

[OAuth 2.0](#_4d34og8) 7

[Authenticate and Authorise](#_2s8eyo1) 7

[Select](#_17dp8vu) 8

[Transfer](#_3rdcrjn) 8

[Information Transfer](#_26in1rg) 8

[Print Job Transfer](#_lnxbz9) 9

[Release](#_35nkun2) 9

[Report](#_1ksv4uv) 10

[Open standards summary](#_44sinio) 10

[**Processes and Workflow**](#_2jxsxqh) **10**

[System Overview](#_z337ya) 10

[Definitions](#_3j2qqm3) 11

[End User Workflow](#_1y810tw) 12

[Technical Workflow](#_4i7ojhp) 12

[OAuth2](#_2xcytpi) 14

[Service Registration](#_1ci93xb) 14

[OAuth Device Authorisation Grant](#_3whwml4) 14

[IPP](#_2bn6wsx) 15

[IPP Infra Flow](#_qsh70q) 16

[IPP Model and Semantics](#_3as4poj) 18

[IPP Operators](#_1pxezwc) 18

[IPP Client Operators](#_49x2ik5) 19

[IPP Proxy Operators](#_2p2csry) 20

[IPP Attributes](#_147n2zr) 21

[Common Attributes](#_3o7alnk) 22

[Job Attributes](#_23ckvvd) 22

[Document Attributes](#_ihv636) 22

[Printer Attributes](#_32hioqz) 23

[IPP Event Notifications and Subscriptions](#_1hmsyys) 24

[IPP Encoding and Transport](#_41mghml) 25

[Other Requirements](#_2grqrue) 25

[Internationalization Requirements](#_vx1227) 25

[Document Format](#_3fwokq0) 26

[**Security**](#_1v1yuxt) **26**

[**Assumptions**](#_4f1mdlm) **26**

[**Limitations**](#_2u6wntf) **26**

[Print Device Capability](#_19c6y18) 26

[**Other Functions**](#_3tbugp1) **27**

[Quotas and Balances](#_28h4qwu) 27

[User Authorisation](#_nmf14n) 27

[Cross-charging](#_37m2jsg) 27

[**Out of Scope**](#_1mrcu09) **28**

[**GovPrint Interoperability Working Group**](#_46r0co2) **28**

[**Appendix A - Definitions**](#_2lwamvv) **28**

# Introduction

Printing is still an important operational service that is required by public sector agencies. Even with the increasing move to digital ways of working the demand for printing although declining will persist for the foreseeable future.

Historically, each agency has delivered their own print service, usually via contracts with suppliers to provide print devices, software and support services, often delivered as a Managed Print Service.

Two major trends are impacting on the long term suitability of this departmental focused approach.

Firstly, the increase in multi-agency buildings, for example the Government Hubs, where multiple organisations work in a single location.

Secondly, the trend for an increasingly mobile workforce, with users working in multiple locations, including locations provided by other agencies. In addition, the increase of inter-agency work means that more users are working in multiple locations.

In addition, the UK Government [Technology Code of Practice](https://www.gov.uk/government/publications/technology-code-of-practice/technology-code-of-practice) states ‘avoid duplicating effort and unnecessary costs by collaborating across government and sharing and reusing technology, data, and services.

Put simply, users need to be able to access and consume print services wherever they are working, much like the way they can access their corporate systems via the internet wherever they are working. A good example of how this has delivered benefits in other areas is GovWifi. GovWifi enables users to roam across multiple locations managed by different agencies accessing internet services using a single identity. Print services should be accessible in the same way.

Currently the only option for meeting this requirement is for each department to provide some form of ‘guest printing’ service, often via printing via email or a web upload service.

This approach has limitations:

* Users require separate guest accounts for each service
* Users may use the same password for their guest account as they use for their departmental systems
* Each service may provide different levels of functionality with a different user experience
* Limited ability to manage who can access the service - the guest users department cannot limit their ability to print and the host department cannot control who has access to the service
* Departments may limit the functions available to guest users

An alternative that is often used in locations where one department is hosting another is for the hosting department to provide additional printers for use by the hosted departments users.

Neither of these solutions are ideal both in terms of usability, efficiency, security and cost.

What other options are available to meet the requirement for print roaming?

One option would be for all departments to use a common print service. There are multiple reasons why this option is unlikely to be acceptable including commercial issues and security risks, plus it would be difficult for the service to meet all agencies individual requirements.

Ideally what is required is the ability for each department to implement a print service that meets their specific needs but can also interoperate with other departments print services, allowing users to roam and access print services in other departments locations.

The aim of this discussion paper is to put forward potential solutions based on open standards that can provide this capability.

GDS are proposing to set up a GovPrint Interoperability Working Group (GPIWP) including a Basecamp project group to support discussion and collaboration. The primary aim of the GPWIP will be to extend and build on this discussion paper and develop an interoperability standard.

For the avoidance of doubt, there is no expectation or requirement for suppliers that participate in the GPIWP to include the functionality defined in the standard in their product(s). The aim will be to allow departments to decide whether to include interoperability as a requirement in their print procurements and based on market demand suppliers can decide whether to support this requirement by implementing the standard.

The aim of the current document is to define a Minimum Viable Product (MVP) that can be used for the development of a demonstration service and a V1 product to be used for an operational service.

# Problem Statement

A typical print service normally includes:

* print devices, including printers and multi-functional devices
* print infrastructure, including print servers
* print software, including software to support ‘follow me’ printing
* print support services, including provision of consumables and support services such as break fix and software support

This discussion paper focuses on how the print software component could enable interoperability between print services using the open standards proposed.

The following summarises the problem GovPrint interoperability seeks to address.

How can:

* a department implement a print service for their locations that meets their specific needs and
* enable their print service to communicate with other departments print services using open standards such that
* users can access print services in locations where print services are provided by other departments (and vice versa)

To illustrate this consider the following scenario:

* Department A has implemented System A
* Department B has implemented System B
* System A can communicate with System B using open standards

This enables the following:

* User A (from Department A) is working in one of Department B’s buildings
* User A submits a print job to System A, their departments print service (the Home Service)
* User A goes to a print device connected to System B provided by Department B (the Guest Service)
* User A is authenticated using their System A credentials and authorises System A to allow send print jobs to System B
* System B receives the print job and sends it to the print device to be printed

# Interoperability Functions

This section outlines the key functions that are required to meet the requirements outlined in the problem statement.

Interoperability between print services requires the following functions to be supported:

* *Authenticate* - authentication of User A using their Home Service credentials when they want to release a print job at a print device connected to a Guest Service
* *Select* - enable User A to select which jobs they want transferred from the Home Service to the Guest Service
* *Authorise* - authorise the Home Service to send / release print jobs to the Guest Service
* *Transfer* - transfer the selected print jobs from the Home Service to the Guest Service
* *Release* - enable User A to select a print device where the transferred job(s) should be printed
* *Report* - transfer reporting information back to the Home Service to confirm the print job has been printed

# Candidate Open Standards

 It is proposed to use two standards to support the interoperability functions outlined above:

* Internet Printing Protocol (IPP)
* OAuth 2.0

## Internet Printing Protocol

IPP is an open standard for printing and provides a secure application level protocol used for network printing. IPP defines an abstract model for printing, including operations with common semantics to communicate between clients and printers and is managed by the Printer Working Group (PWG).

IPP uses HTTP as its transport protocol and can support TLS (HTTPS). Each IPP request is an HTTP POST and the corresponding IPP response is returned in the POST response. HTTP POST requests can also be authenticated using any of the usual HTTP mechanisms such as OAuth2 and certificates.

IPP messages use a common format for requests from the IPP Client to the IPP Printer and responses from the IPP Printer to the IPP Client.

IPP supports a set of operations and attributes that are defined in a range of standards published by the Printer Working Group (PWG) the current proposal is based on the following standards:

* IPP Version 2.0, 2.1, and 2.2 (PWG 5100.12)
* IPP Shared Infrastructure Extensions (INFRA) (PWG 5100.18).
* IPP Event Notifications and Subscriptions (RFC 3995)
* Ippget Delivery Method for Event Notifications (RFC 3996)
* The Cloud Imaging Requirements and Model [PWG5109.1]

## OAuth 2.0

OAuth2 is an open standard for access delegation, commonly used as a way for users to grant a service access to their information held on another service. OAuth2 is an authorisation framework that enables applications to obtain limited access to user's resources on an HTTP service such as IPP. The specification and associated RFCs are developed by the IETF OAuth Working Group.

 It works by delegating user authentication to an authorization service which is linked to the user’s departmental identity management service. OAuth2 supports a number of Grant Types that can be used for different use cases and it is proposed to use Device Authorisation Grant to enable the Guest Service to obtain an Access Token from the Home Service. This Access Token can then be used by the Guest Service to securely retrieve print jobs from the Home Service.

Oauth2 supports a registration process enabling a Guest Service to register with a Home Service to access the OAuth service. This process establishes the credentials for identifying and authenticating the Guest Service when it requests Access Tokens.

## Authenticate and Authorise

Authentication and authorisation will be considered together as the proposed standards support both functions.

It is proposed to use *OAuth 2.0* (OAuth2) Device Authorisation Grant to support authentication and authorisation. A full description of OAuth2 is outside the scope of this discussion paper and the focus will be on how OAuth2 Device Authorisation Grant can be applied to support authentication and authorisation. The OAuth2 Device Authorisation Grant enables the Guest Service to obtain an Access Token from the Home Service to be used when requesting print jobs using IPP..

Key discussion and decisions:

* are there alternatives to OAuth2 that should be considered?
* are there alternative OAuth2 flows that should be considered e.g. Open ID Connect?

## Select

The user selects jobs in an app or web page that is provided by their Home Service via an end user device e.g. laptop / mobile.

No specific open standards are required for this functionality as it will be implemented within the Home Service end user app / web page.

## Transfer

Two types of information need to be transferred between interoperating systems:

* information about the print jobs
* print jobs

### Information Transfer

IPP defines a transport mechanism (https) and a semantics model that can support the transfer of both types of information between the interoperating services. This includes a set of operators (functions), objects such as printer, document and job and a wide range of attributes for each type of object.

Key discussion and decisions:

* can IPP provide a secure method to transfer information?
* can IPP provide the range of operators, objects and attributes required to support interoperability?

### Print Job Transfer

The transfer of print jobs requires three elements:

* a mechanism to transfer the print job
* a standard format for a print job that can be supported by most print services / print devices
* a mechanism to define attributes of the print job e.g. double sided output

It is proposed to transfer print jobs using IPP using the IPP Shared Infrastructure Extensions (IPP INFRA) model. This option enables the Guest Service to request print jobs from the Home Service using the Access Token granted via OAuth2.

It is proposed to transfer the print jobs in formats recommended within the IPP Everywhere standard:

* PDF
* JPEG
* PWG Raster Format
* Open XML Paper Specification

It is proposed to use IPP Job / Document Attributes to define attributes of the print job e.g. double sided printing, where these are not defined within the job format.

Key discussion and decisions:

* are there alternatives to IPP INFRA (or IPP) that should be considered?
* what file formats should be supported e.g. all formats defined by IPP Everywhere or PDF only?
* for file formats selected what versions should be supported e.g. PDF 1.7 as defined by IPP Everywhere?
* what Job / Document Attributes are required?

## Release

Print job(s) will be released at the print device where the user initiated the OAuth2 Device Authorisation Grant flow. No specific open standards are required for this functionality and management of this process will be the responsibility of the Guest Service.

## Report

The Guest Service releasing the print job must provide feedback to the Home Service confirming the print job has been printer (or not). The feedback must also include information about where and when it was printed plus a relevant print volume metric.

It is proposed to use IPP Infra together with the relevant job / document attributes to enable the Guest Service to inform the Home Service of the outcome.

Key discussion and decisions:

* are there alternatives to IPP INFRA (or IPP) that should be considered?
* what Job / Document Attributes are required?

## Open standards summary

The following table summarises the proposed open standards that can be used to provide the required interoperability functions.

Table 1 - Proposed interoperability open standards

|  |  |
| --- | --- |
| **Function** | **Open Standard** |
| Authenticate and Authorise | OAuth 2.0 Device Authorisation Grant |
| Transfer print job information | IPP / IPP Infra |
| Transfer print job | IPP / IPP Infra |
| Print job format | PDFIPP Job Attributes |
| Report | IPP Job / Document Attributes |

# Processes and Workflow

## System Overview

Diagram 1 provides a summary of the main components of the print systems including the IPP and OAuth2 components required to support interoperability.

Diagram 1 - Interoperability Service Overview



## Definitions

* Home Service - the print service operated by the user’s home department
* Guest Service - a print service operated by another department
* Each service comprises the following components
	+ Client - software installed on user devices and used to submit print jobs
	+ System - software that can accept, store and release print jobs
		- System A - system component of Home Service
		- System B - system component of Guest Service
	+ Printer - output device
* IPP Client - component of System A that initiates connections and sends Print Jobs to an Infrastructure Printer
* Infrastructure Printer - component of System A that stores print jobs submitted by IPP Client, informs System B IPP Proxy that jobs are ready and transfers Print Jobs to System B IPP Proxy when requested
* IPP Proxy - component of System B that subscribes to notifications from one or more Infrastructure Printers and retrieves print jobs when they are made available by an Infrastructure Printer
* IPP Printer - component of System B that listens for connections from IPP Proxy and receives Print Jobs from IPP Proxy and forwards to a Target Printer
* Target Printer - the output device where the user initiated the Guest Print request

## End User Workflow

Standard End User Print Workflow when printing on the user’s Home Service (shown in green Diagram 1):

1. User submits Print Job via Client to Input Queue
2. Pull Print -
	1. Print Job held in Storage
	2. User selects Printer for job release and Print Job sent via Output Queue to Printer
3. Push Print
	1. Print Job sent direct to Printer via the Output Queue

Interoperability End User Print Workflow when printing on the Guest Service (shown in green / red / blue Diagram 1):

1. On Home Service User submits Print Job via Client to Pull Print Input Queue
2. Print Job held in Storage (Push Print)
3. At Print Device on a Guest Service the User selects ‘guest printing’ and then selects the name of their Home Service on the control panel of the device
4. On the Home Service the User authenticates and selects stored Print Jobs for release on Guest Service
5. Home Service sends selected Print Jobs to the Guest Service which releases the jobs on the Target Printer

## Technical Workflow

Diagram 2 - Technical Workflow



The following provides a high level summary of the technical workflow when the User is printing on a Guest Service:

1. The User submits print jobs to a pull print queue and these are stored on the Home Service
2. User selects ‘guest print’ on a print device connected to the ‘Guest Service’ and confirms their Home Service name, this starts OAuth2 Device Authorisation Grant flow
3. Guest Service requests a ‘device code’ from the Home Service OAuth service and displays this on the Print Device
4. On their Home Service the User:
	1. Authenticates via an App or Webpage
	2. Selects the stored print jobs they want to release
	3. Enters the ‘device code’
	4. Authorises the Home Service to send print jobs to the Guest Service
5. The Home Service OAuth2 service provides an Access Token to the Guest Service to use when retrieving the print jobs from the Home Service
6. The Guest Service requests print jobs from Home Service and presents the Access Token. The Home Service checks the validity of the Access Token and if valid releases the print jobs for the user that authorised the Access Token
7. The Guest Service IPP Printer releases the print jobs into the queue for the Target Printer where the User initiated guest printing

## OAuth2

### Service Registration

The Home Service will need to register with each Guest Service it will wish to interoperate with. This process will follow the typical process for registering an application with an OAuth2 service provider. The registration process will create client credentials and register redirect URLs.

### OAuth Device Authorisation Grant

Table 2 outlines the OAuth2 flow required to enable the Home Service to issue an Access Token to the Guest Service following user authentication on a Home Service device.

Table 2 - OAuth2 Device Authorisation Grant flow

|  |  |
| --- | --- |
| **End User Workflow** | **OAuth Device Grant** |
| User selects ‘Guest Print’ and confirms their Home Service on the control panel of a printer connected to a Guest Service (Target Printer) | OAuth Device Grant flow begins |
|  | Guest Service requests a ‘Device Code’ from the Home Service OAuth2 service using the URL and credentials configured during OAuth2 service registration |
| Home Service OAuth2 service responds with a JSON payload with:‘device\_code’‘user\_code’‘expires\_in’‘verification URL’ is normally provided but is not used as the user is accessing the authorisation service from their Home Service |
| Guest Service displays ‘user\_code’ value on the control panel of the Target Printer (no requirement to display verification URL) |
| Guest Service starts polling the Home Service OAuth2 service to request an Access Token, and continues until End User grants or denies the request or the device\_code expires (expires\_in) |
| User authenticates by logging into Home Service on webpage or Mobile App |  |
| User selects the Print Jobs they want to release |
| User enters ‘user code’ and confirms they want to authorise the release of print jobs at the specific device |
|  | Home Service OAuth2 service sends ‘Access Token’ to Guest Serviceif the User does not grant access an ‘access\_denied’ error is returned to the Guest ServiceIf device code expires an ‘expired\_token’ error is returned to the Guest Service  |
| ‘Guest Service’ can begin IPP Infra process using ‘Access Token’ for IPP authentication when fetching jobs / documents (see IPP section below) |

## IPP

As outlined previously IPP defines a wide range of operators, objects and attributes, the current document defines the MVP/V1 requirements and many of the typical functions of IPP are not be required (it is likely that the range of functions supported will be aligned to IPP/2.0 conformance level (PWG 5100.12) and IPP Infrastructure Extensions (PWG 5100.18)). Other functions required to provide a more feature rich implementation may be added in future revisions.

IPP Infrastructure Extensions (IPP Infra) defines an extended model based on IPP that enables a print service to request print jobs from another print service. This model comprises four components:

* IPP Client
* Infrastructure Printer
* IPP Proxy
* IPP Printer

In order to support IPP Infra, the IPP Client, Infrastructure Printer, and IPP Proxy must use a common AAA Framework, in this case it is proposed to use OAuth 2.0.

### IPP Infra Flow

Table 3 outlines the IPP Infra process flow required to enable the Guest Service to retrieve print files from the Home Service following the successful completion of the OAuth2 Device Authorisation Grant flow.

Table 3 - IPP Infra flow

|  |  |  |  |
| --- | --- | --- | --- |
| **IPP Client** | **Infrastructure Printer** | **IPP Proxy** | **IPP Printer** |
|  |  | IPP Proxy creates subscription on Infrastructure Printer at service startup |  |
| IPP Proxy polls Infrastructure Printer for fetchable job events |
| IPP Client sends selected Print Jobs to Infrastructure Printer including ‘job-originating-user-name’ = UPN |  |
|  | Infrastructure Printer updates ‘job-fetchable’ event  |
|  | IPP Proxy requests details of fetchable jobs and presents Access Token |
| Infrastructure Printer checks validity of Access Token with OAuth2 service and gets the ‘username’ associated with the token |  |
| Infrastructure Printer returns details of fetchable jobs where ‘job-originating-user-name’ = ‘username’ |
|  | IPP Proxy creates job on IPP Printer with ‘output-device-uuid’ = OAuth2 flow start device name |
|  | Print Job created with output device set to device where the OAuth flow was initiated |
| IPP Proxy sends an acknowledgement to Infrastructure Printer |  |
| IPP Proxy requests document(s) associated with the Print Job using Access Token |
| Infrastructure Printer validates Access Token |  |
| Infrastructure Printer returns document(s) to IPP Proxy |
|  | IPP Proxy adds document(s) to Print Job on IPP Printer |
|  | Documents added to Print Job |
| IPP Proxy sends an acknowledgement to Infrastructure Printer |  |
|  | IPP Printer forwards Print Job to output queue linked to output-device-uuid |
| IPP Printer sends update to IPP Proxy confirming Job completed |
| IPP Proxy sends update to Infrastructure Printer confirming Job completed |  |
| Infrastructure Printer sends update to IPP Client confirming Job completed |  |
| Job Completed |  |
|  | IPP Proxy continues to poll Infrastructure Printer for fetchable job events |

### IPP Model and Semantics

IPP defines a wide range of Operators (actions) and object Attributes (information). To support interoperability a limited set of Operators and Attributes are required and these are set out in tables 4 to 9 below. Although a limited range of Operators and Attributes are required for the MVP version this will expand in future versions to provided enhanced functionality (it is expected that the service will support the Operators and Attributes defined in IPP/2.0 (PWG 5100.12)).

#### IPP Operators

IPP supports a wide range of operators but for the MVP solution only the critical functions required to transfer Print Jobs from the Home Service to the Guest Service are defined. The following tables define the key operators that MUST be supported by each component to deliver interoperability for the MVP / V1 standard.

##### IPP Client Operators

The key function of the selected IPP Client Operators is to:

1. create a job on the Infrastructure Printer
2. submit documents to be included in the job

All communication between the IPP Proxy and Infrastructure printer must support TLS (uri-security-supported = tls) and basic authentication (uri-authentication-supported = basic) for the MVP.

Table 4. IPP Client Operators Required for MVP

|  |  |  |
| --- | --- | --- |
| **Operator1** | **Key Attributes2** | **Comments** |
| Create-Job**3** (IPP C>IP) | job-idjob-originating-user-name | Create a Job on the Infrastructure Printer with job-originating-user-name = UPN |
| Create-Job Response4 (IP>IPP C) | job-id |  |
| Send-Document(IPP C>IP) | document-namedocument-formatcopiesmediasidesdocument-access<document data> | Send each document to be included in the Job i.e. all documents selected by the User. MVP will support minimum output format options.document-access attribute must provide authentication information for the referenced document. |
| Send-Document-Response(IP>IPP C) | job-iddocument-name |  |
| Cancel-Job (IPP C>IP) | job-id | Cancel specified job and associated documents |
| Cancel-Job (IP>IPP C) |  |  |

Notes:

1. Only operators that are required to support interoperability are listed, other operators defined as REQUIRED in IPP/2.0 and IPP Infra must be supported.
2. Only attributes that are required to support Interoperability are listed, other attributes defined as REQUIRED in IPP/2.0 and IPP Infra must be supported.
3. It is proposed to only support Print-Job only in MVP/V1 but it may be worthwhile evaluating the benefits of using Create-Job / Send-Document operators
4. Whenever an Operator is used the target of the operator must return the relevant ‘xxx-state’, ‘xxx-state-message’ and ‘xxx-state-reasons’ in the operator response

##### IPP Proxy Operators

The key function of the selected IPP Proxy Operators is to:

1. Create a printer subscription on the Infrastructure Printer
2. Get notifications
3. Retrieve fetchable print jobs and associated documents

All communication between the IPP Proxy and Infrastructure printer must support TLS (uri-security-supported = tls) and basic authentication (uri-authentication-supported = basic) for the MVP.

Table 5 - IPP Proxy Operations Required for MVP

|  |  |  |
| --- | --- | --- |
| **Operator1** | **Attributes2** | **Comments** |
| Create-Printer-Subscriptions(IPP P>IP) | notify-events | Subscription created for specific events relevant to fetchable jobs |
| Create-Printer-Subscriptions response(IP>IPP P) |  |  |
| Renew-Subscription(IPP P>IP) |  |  |
| Cancel-Subscription(IPP P>IP) |  |  |
| Get-Notifications request(IPP P>IP) | notify-subscription-ids |  |
| Get-Notifications response(IP>IPP P) |  | Returned job-fetchable event notifications |
| Get-Jobs request(IPP P>IP) | job-state-reasons = ‘job fetchable’access-oauth-tokenaccess-oauth-uri | Access Token included in Get-Jobs request |
| Get-Jobs response(IP>IPP P) |  | Jobs with job state reason set to fetchable and ‘job-originating-user-name’ = ‘username’ |
| Fetch-Job request(IPP P>IP) | job-id |  |
| Fetch-Job response(IP>IPP P) |  |  |
| Acknowledge-Job(IPPP>IP) |  |  |
| Fetch-Document(IPP P>IP) | job-iddocument-numberaccess-oauth-tokenAccess-oauth-uri | Fetch request includes the Access Token obtained via OAuthJob created on IPP Printer has output-device-uuid = OAuth2 start device |
| Fetch-Document Response(IP>IPP P) | status-codestatus-messagedetailed-status-message<document data> | File to be printed if correct Access Token is presented |
| Acknowledge-Document(IPP P>IP) | printer-urijob-idoutput-device-uuid requesting-user-name | Confirms document has been accepted by IPP Printer |
| Update-Document-Status |  |  |
| Update-Job-Status |  |  |

#### IPP Attributes

Tables 6 to 9 define the key attributes required to support interoperability for the MVP / V1 standard. It is important to note that this list only includes attributes that are critical to interoperability only and does not include all MANDATORY attributes as defined in IPP/2.0 and IPP Infra. Suppliers MUST assume that all REQUIRED IPP/2.0 and IPP Infra attributes must be supported (even if they are not used to support interoperability).

All objects must support relevant xxx-status-code, xxx-status-message and xxx-detailed-status-message and if applicable xxx-state, xxx-state-message and xxx-state-reasons.

##### Common Attributes

Table 6 - Common attributes used by all operators

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Operator(s)** | **Defaults / Comments** |
| attributes-charset | Included with all operators |  |
| attributes-natural-language | Included with all operators |  |
| printer-uri | Included with all operators | Printer naming conventions when used when defining connections between interoperating services e.g. https://deptA.gov.uk:631/ipp/int\_print |
| requesting-user-name | Included with all operators | User’s UPN |

##### Job Attributes

Table 7 - Job Attributes

|  |  |
| --- | --- |
| **Attribute** | **Defaults / Comments** |
| job-id | MVP  |
| job-name | MVP - Unique value generated by System A, can be used for reporting and audit purposes. This SHOULD NOT be the document name used in System A as this could include sensitive information. |
| job-state | MVP |
| job-state-reasons | MVP - set to ‘job-fetchable’ for jobs that Guest Service can retrieve |
| output-device-uuid | MVP - IPP target printer where OAuth2 Device Grant flow was initiated |

##### Document Attributes

Table 8 - Document Attributes

|  |  |
| --- | --- |
| **Attribute** | **Defaults / Comments** |
| document-format | MVP ‘application/pdf’ |
| document-name | MVP this should be a system generated value and not represent the source document name e.g. document.docx |
| document-number | MVP |
| document-job-id | MVP |
| copies | MVP |
| sides | MVP |
| finishings | V1 |
| media | MVP |
| number-up | V1 |
| impressions | MVP |
| date-time-at-created | V1 |
| date-time-at-completed | MVP |
| impressions-completed | MVP, this attribute does not distinguish between colour and mono impressions |
| job-media-sheets-completed | V1 |
| fetch-status-code | MVP |
| fetch-status-message | MVP |
| access-oauth-token | MVP - Access Token used to get / fetch job / document |
| access-oauth-uri | MVP - uri of OAuth server that issued Access Token, this will be the same OAuth server used by the printer |
| output-device-uuid | MVP - IPP target printer |

##### Printer Attributes

Table 9 - Printer Attributes

|  |  |
| --- | --- |
| **Attribute** | **Defaults / Comments** |
| printer-name | MVP |
| uri-authentication-supported | MVP - value set to basicV1 - value set to certificate |
| uri-security-supported | MVP tls |
| ipp-versions-supported | MVP 1.1 |
| document-format-default | MVP application/octet-stream |
| document-format-supported | MVP application/pdf |
| operations-supported | MVP Create-Job, Send-Document and Cancel-Job |
| copies-default | V1 |
| copies-supported | V1 |
| sides-default | V1 |
| sides-supported | V1 |
| finishings-default | V1 |
| finishings-supported | V1 |
| media-default | V1 |
| media-supported | V1 |
| Document-access-supported1, 2 | MVP - attribute must include ‘access-oauth-token’ |
| oauth-authorization-server-uri | MVP - OAuth authorisation server used by the Infrastructure printer |
| ipp-features-supported | MVP - infrastructure printers must include 'infrastructure-printer' in this attribute |

Notes:

1. The Infrastructure Printer must support ‘document-access-supported’ / ‘access-oauth-token’
2. IPP supports a range of authentication methods in addition to OAuth2 including username/password, kerberos and certificates, this would enable extension of the service to support additional methods

#### IPP Event Notifications and Subscriptions

Infrastructure Printers MUST conform to the IPP Event Notifications and Subscriptions [RFC3995] and the 'ippget' Delivery Method for Event Notifications [RFC3996].

Infrastructure Printers MUST support the 'job-completed', 'job-config-changed', 'job-created', 'job-progress', 'job-state-changed', 'job-stopped', 'printer-config-changed', 'printer-queue-order-changed', 'printer-state-changed', and 'printer-stopped' events.

Table 10 - Event Notifications (events subscribed to by the IPP Proxy)

|  |  |
| --- | --- |
| **Event (notify-events)** | **Defaults / Comments** |
| job-fetchable | MVP |
| document-fetchable | MVP |
| document-completed | MVP |

Table 11 - Event Attributes

|  |  |
| --- | --- |
| **Attribute** | **Defaults / Comments** |
| document-fetchable | MVP |
| job-impressions-completed | MVP |

### IPP Encoding and Transport

This section outlines key standards to be supported but is not a full description of these standards.

The IPP Client and IPP Printer must:

* Support the complete HTTP/1.1 protocol as defined in [RFC7230] [RFC7231] [RFC7232] [RFC7233] [RFC7234] [RFC7235]
* Support chunking as defined in section 4.1 of [RFC7230;
* Support HTTPS with mutual authentication using certificates [RFC2818]
* Conform to the Transport Layer Security (TLS) Version 1.2 [RFC5246] or higher
* Support NCSC TLS Foundation Profile

Authentication and Authorisation of IPP Client connection and submission of request to IPP Printer must support:

* MVP - IPP ‘oauth’ authentication method

### Other Requirements

#### Internationalization Requirements

It is recommended that the MVP / V1 standard supports:

1. The Universal Character Set (UCS) Transformation Format -- 8 bit (UTF-8) [STD63] encoding of Unicode [UNICODE] [ISO10646]; and
2. The Unicode Format for Network Interchange [RFC5198] that requires transmission of well-formed UTF-8 strings and recommends transmission of normalized UTF-8 strings in Normalization Form C (NFC) [UAX15].

## Document Format

It is recommended to limit the format of the print files that can be transferred to PDF files that meet the ISO 32000-1:2008 (PDF 1.7) standard.

# Security

Where appropriate communication between services must support the following:

* TLS v1.2 for all connections and communications between print services
* mutual authenticate service endpoints using X.509 certificate validation

# Assumptions

None currently documented.

# Limitations

## Print Device Capability

Print devices support a range of functions e.g. colour / mono, paper size, finishing options etc. A user may submit a job as duplex and colour but release it at a device that does not provide this capability e.g. simplex and mono.

The simplest solution is for the print device to make ‘best efforts’, e.g. in the case above the device prints the duplex colour as simplex mono. This may require defining some basic rules such as those outlined in Table 2 Compatibility translations.

For the MVP / V1 versions of the interoperability standard only limited job and document attributes will be supported.

Table 12 - Compatibility translations

|  |  |  |
| --- | --- | --- |
| **Submitted Requirement** | **Device Capability** | **Output** |
| Colour | Mono | Mono |
| Duplex | Simplex | Simplex |
| Staple | No Staple | No Staple |
| Punch | No Punch | No Punch |
| A3 | A4 | A4 |

# Other Functions

## Quotas and Balances

If the home services uses quotas / balances to manage print activity, these should be checked before the print job is sent from the home service to the guest service.

If the guest service allows print job attributes to be changed e.g. number of copies, the final report back from the guest to the home must report what was actually printed.

## User Authorisation

Departments may wish to control which users can access guest print services i.e. control which of their users can access guest printing on other services. When a user authenticates at a Guest service the Home service should be able to check whether a user has been granted the right to access guest printing before approving access to the service and providing the Guest Service with an Access Token.

Departments may wish to set quotas for the volume of print users can release at a guest service. This should be set in the home service and the home service should check this before authorising the transfer of the print job(s) to the guest service.

## Cross-charging

The mechanism for cross-charging is out of scope.

# Out of Scope

TBC

# GovPrint Interoperability Working Group

GDS propose to setup a GovPrint Interoperability Working Group (GPIWG). The group will comprise interested suppliers and government departments. The primary aim of the group will be to define a standard for interoperability. There would be no requirement for suppliers to implement the standard within their solutions at this stage. The standard would not be integrated into products until the market demand is clearly stated e.g. in future procurement tenders. This approach is very similar to the way the IEEE PWG operate in that they define the standard and leave it to suppliers to decide whether to implement the standard based on market demands.

Taking this approach would mean that suppliers do not at this stage need to invest significant resources, they simply need to participate in reviewing and commenting on the proposed standard. The aim will be to produce a technical standard for interoperability that everyone considers meets the users requirements and can be supported by the suppliers software.

# Appendix A - Definitions

OAuth 2.0 -

OAuth 2.0 (OAuth) is an open standard for access delegation, it provides to clients a "secure delegated access" to server resources on behalf of a resource owner. It specifies a process for resource owners to authorize third-party access to their server resources without sharing their credentials. (<https://en.wikipedia.org/wiki/OAuth>)

OAuth Device Authorisation Grant -

OAuth Device Authorisation Grant is an authorisation layer enabling authentication and authorisation of a user via a secondary device such as a mobile app. OAuth Device Grant is currently a draft standard.

IPP / IPPS / IPP INFRA -

The Internet Printing Protocol (IPP) is an internet printing protocol for communication between client devices and printers or print servers. IPP also supports access control, authentication, and encryption (IPPS). IPP Infrastructure Extensions (IPP INFRA) enable print devices or print servers to request print jobs from clients. (<https://en.wikipedia.org/wiki/Internet_Printing_Protocol>)

PDF -

Portable Document Format (PDF) is a file format developed by Adobe to present documents, including text formatting and images, in a manner independent of application software, hardware, and operating systems and is based on the PostScript language. PDF was standardized as an open format, ISO 32000. (<https://en.wikipedia.org/wiki/PDF>)

JSON -

JavaScript Object Notation (JSON) is an open-standard file format that uses human-readable text to transmit data objects consisting of attribute–value pairs and array data types (or any other serializable value) (<https://en.wikipedia.org/wiki/JSON>)