[MS-PRSOD]:
Print Services Protocols Overview

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.

- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.

- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.

- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft Open Specification Promise or the Community Promise. If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.

- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.

- **Fictitious Names.** The example companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

This document provides an overview of the Print Services Protocols Overview Protocol Family. It is intended for use in conjunction with the Microsoft Protocol Technical Documents, publicly available standard specifications, network programming art, and Microsoft Windows distributed systems concepts. It assumes that the reader is either familiar with the aforementioned material or has immediate access to it.

A Protocol System Document does not require the use of Microsoft programming tools or programming environments in order to implement the Protocols in the System. Developers who have access to Microsoft programming tools and environments are free to take advantage of them.

Abstract

This document describes a distributed system of print servers that manage printers and make them available to print clients. One or more servers may be used, each server independently managing one or more printers. Print clients use the component protocols to submit print jobs, manage jobs, receive job notifications, and administer printer drivers and print queues.

This document describes the functionality of the Print Services System and how the protocols in this system interact, namely [MS-CIFS], [MS-EMFSPOOL], [MS-FSCC], [MS-GPDPC], [MS-PAN], [MS-PAR], [MS-RAP], [MS-RPRN], [MS-SMB], [MS-SMB2], and [MS-WPRN]. It does not restate the processing rules and other details that are specific for each protocol; these details are described in the protocol specifications for each of the protocols and data structures that make up this system. It covers client/server scenarios, and does not address functionality that the Print Services System can provide to manage a single printer attached to a single computer. It provides examples of common user scenarios.

Revision Summary

Date	Revision History	Revision Class	Comments
09/23/2011	1.0	New	Released new document.
12/16/2011	1.0	No change	No changes to the meaning, language, or formatting of the technical content.
03/30/2012	2.0	Major	Significantly changed the technical content.
07/12/2012	3.0	Major	Significantly changed the technical content.
10/25/2012	3.0	No change	No changes to the meaning, language, or formatting of the technical content.
01/31/2013	3.0	No change	No changes to the meaning, language, or formatting of the technical content.
08/08/2013	4.0	Major	Significantly changed the technical content.
11/14/2013	4.0	No change	No changes to the meaning, language, or formatting of the technical content.
02/13/2014	4.0	No change	No changes to the meaning, language, or formatting of the technical content.

Contents

1 Introduction

This document describes the Print Services System (PSS), which supports the use and management of a distributed print infrastructure. It describes how the Print Services System can be used in workgroups and **domain**-based networks, and how print servers and print clients can use the member protocols of the system to share one or more printers between one or more print clients.

The Print Services System can serve workgroups, where printers are shared among individual clients, and domain-based networks, where there can be multiple print servers deployed in a cluster configuration, with the print client configuration managed through the **Active Directory** System. The differences between these use scenarios are those described in the protocols described later in this document.

1.1 Conceptual Overview

This section provides an overview of the following concepts, which are required for understanding this document.

- **Print spoolers**

- Print spooler services

- Print queues

- Printers and print data formats

- **Printer drivers** and **print processors**

Familiarity with the Windows **Graphics Device Interface (GDI)**, documented in [MSDN-WindowsGDI], is also necessary for understanding this document.

1.1.1 Print Spoolers

A print spooler running on a print client exposes local printing APIs to applications, receives the print output of an application, and sends it to a shared print queue on a print server. The print output is either a data stream in a format that can be processed on a specific printer, or a generic metafile that can be transcoded and/or re-rendered into a format that can be processed on a specific printer.

A print spooler running on a print server shares print queues to expose them for use by print clients. The print server is responsible for any necessary transcoding and/or re-rendering of a print output data stream received from a client before the print spooler buffers the data stream and sends it to a target printer.

On both a print server and print client, the print spooler manages printer driver components and other components that aid in the creation of application print output and translation of metafiles, and allows applications to obtain metrics and status information about printers.

The Windows implementations of the print client and print server roles are provided by the print spooler component. Each Windows system runs a print spooler. Therefore, each Windows computer can act as a print client or a print server.

1.1.2 Print Spooler Service

The print spooler service is a service running on each computer participating in the Print Services System. The print spooler service implements the print client and print server roles, allowing each participating system to act as a print client, administrative client, or print server for the Print Services System.

Implementation of the print client role can include implementation of the print server role in the print spooler service due to the dual client/server nature of the protocol described in [MS-RPRN].

If an implementation acts as print client only, it supports all methods described in [MS-RPRN] and can optionally also support the Print System Asynchronous Remote Protocol [MS-PAR].

For the print server role, the print spooler service registers the **RPC endpoints** for the protocols described in [MS-PAR], [MS-RPRN] and [MS-PAN].

For the print server role, the print spooler service exposes local interfaces that extend Internet Information Services (IIS) to support the Internet Printing Protocol (IPP) described in [RFC2910] and [RFC2911], and the Web Point-and-Print Protocol [MS-WPRN], if configured to support IPP.

For the print client role, the print spooler service can use polling in order to collect printer or **print job** related changes happening on the server or can implement the Print System Remote Protocol [MS-RPRN] notification callback functions exposed in the Print System Remote Protocol [MS-RPRN] RPC endpoint. Firewall settings or Windows Group Policy settings can prevent the client from receiving the Print System Remote Protocol [MS-RPRN] notifications. The Windows behavior is to use polling in such scenarios. Additionally, the print spooler service exposes local interfaces used by client applications to print, obtain print queue status, administer print queues, or perform other print-specific actions.

1.1.3 Print Queues

A print queue is an abstract component residing on a print server to which print jobs are submitted. In many component protocol documents, the phrase print queue is used interchangeably with the word printer. However, in this system document, print queue always refers to the abstract component, and printer refers to the printers.

1.1.4 Printers and Print Data Formats

A printer is a physical **device** provided by an **Independent Hardware Vendor (IHV)** that outputs content on a variety of mediums and encompasses both traditional 2D printers as well as 3D printers, cutters, and CNC milling devices (collectively referred to as 3D manufacturing devices), which output physical objects. A printer consumes a data stream representing a description of a document to be printed or a model to be outputted, in the case of a 3D manufacturing device. The data format of that data stream is defined or selected by the IHV that designed the printer or 3D manufacturing device.

There are several de facto standards for these vendor-defined **page description languages (PDLs)**, including **PostScript**, Portable Document Format [ISO-32000-1], Hewlett-Packard (HP) **Printer Control Language (PCL)**, and the XML Paper Specification (see [MSDN-XMLP]). Other vendor-specific print data formats are in use as well.

In order for applications to use different printer or 3D manufacturing technologies in a uniform manner, the Windows graphical subsystem abstracts the details of the print data formats. Windows relies on the services provided by printer drivers and print processors to abstract details of data formats. Data formats can be native PDLs that can be sent directly to the device, or formats such as

EMF (see [MS-EMFSPOOL]) or the XML Paper Specification (see [MSDN-XMLP]), which have to be re-rendered to a PDL before being sent to a printer.

Regardless of the data format, all print data streams sent from a print client to a print server are collectively referred to as payloads. Print payloads are sent from a print client to a print server by using one of the member protocols of the Print Services System.

1.1.5 Printer Drivers and Print Processors

Printer drivers and print processors are printer-specific and implement a set of functions that Windows calls to perform the following tasks.

- Convert application document or 3D model creation and drawing commands (including drawing of text or 3D object) into one of the data formats that can be processed by the respective printer.

- Provide Windows with information about metrics and capabilities of the printer, such as physical paper or output area dimensions, color capabilities, quality options, paper path options, duplexing options, and so on.

- Provide user interface functionality enabling users to configure settings and default behavior for a printer.

1.2 Glossary

The following terms are defined in [MS-GLOS]:

> **access control list (ACL)**
> **Active Directory**
> **Authentication (1)**
> **device**
> **domain**
> **Domain Name System (DNS)**
> **driver package**
> **driver store**
> **endpoint (2)**
> **enhanced metafile format (EMF)**
> **enhanced metafile spool format (EMFSPOOL)**
> **Graphics Device Interface (GDI)**
> **Internet Protocol security (IPsec)**
> **page description language (PDL)**
> **PostScript**
> **print job**
> **Printer Control Language (PCL)**
> **printer driver**
> **registry**
> **remote procedure call (RPC)**
> **security descriptor**
> **security provider**

The following terms are defined in [MS-PAN]:

> **authenticated user identity**

The following terms are defined in [MS-RPRN]:

branch office print mode
branch office print remote logging
branch office printing
language monitor
port monitor
print processor

The following terms are specific to this document:

deployed printer connection: A computer connection or user connection as described in [MS-GPDPC].

Independent Hardware Vendor (IHV): In the context of this document, an **IHV** is a printer manufacturer, such as Canon or Hewlett-Packard.

print spooler: The component (service) that implements the Print Services System on Windows print clients and print servers. The spooler buffers and orders **print jobs**, as well as converts **print job** data to printer-specific formats.

1.3   References

References to Microsoft Open Specification documents do not include a publishing year because links are to the latest version of the documents, which are updated frequently. References to other documents include a publishing year when one is available.

[C706] The Open Group, "DCE 1.1: Remote Procedure Call", C706, August 1997, https://www2.opengroup.org/ogsys/catalog/c706

[ISO-32000-1] ISO, "Document management -- Portable document format -- Part 1: PDF 1.7", ISO 32000-1:2800, July 2008, http://www.iso.org/iso/catalogue_detail.htm?csnumber=51502

Note   There is a charge to download the specification.

[MS-ADLS] Microsoft Corporation, "Active Directory Lightweight Directory Services Schema".

[MS-ADSC] Microsoft Corporation, "Active Directory Schema Classes".

[MS-ADOD] Microsoft Corporation, "Active Directory Protocols Overview".

[MS-AUTHSOD] Microsoft Corporation, "Authentication Services Protocols Overview".

[MS-BRWS] Microsoft Corporation, "Common Internet File System (CIFS) Browser Protocol".

[MS-CIFS] Microsoft Corporation, "Common Internet File System (CIFS) Protocol".

[MS-DRSR] Microsoft Corporation, "Directory Replication Service (DRS) Remote Protocol".

[MS-EMFSPOOL] Microsoft Corporation, "Enhanced Metafile Spool Format".

[MS-FSCC] Microsoft Corporation, "File System Control Codes".

[MS-FASOD] Microsoft Corporation, "File Access Services Protocols Overview".

[MS-GLOS] Microsoft Corporation, "Windows Protocols Master Glossary".

[MS-GPDPC] Microsoft Corporation, "Group Policy: Deployed Printer Connections Extension".

[MS-GPOL] Microsoft Corporation, "Group Policy: Core Protocol".

[MS-GPOD] Microsoft Corporation, "Group Policy Protocols Overview".

[MS-NRPC] Microsoft Corporation, "Netlogon Remote Protocol".

[MS-PAN] Microsoft Corporation, "Print System Asynchronous Notification Protocol".

[MS-PAR] Microsoft Corporation, "Print System Asynchronous Remote Protocol".

[MS-RAP] Microsoft Corporation, "Remote Administration Protocol".

[MS-RPCE] Microsoft Corporation, "Remote Procedure Call Protocol Extensions".

[MS-RPRN] Microsoft Corporation, "Print System Remote Protocol".

[MS-SMB] Microsoft Corporation, "Server Message Block (SMB) Protocol".

[MS-SMB2] Microsoft Corporation, "Server Message Block (SMB) Protocol Versions 2 and 3".

[MS-SPNG] Microsoft Corporation, "Simple and Protected GSS-API Negotiation Mechanism (SPNEGO) Extension".

[MS-WPRN] Microsoft Corporation, "Web Point-and-Print Protocol".

[MS-WPO] Microsoft Corporation, "Windows Protocols Overview".

[MS-WUSP] Microsoft Corporation, "Windows Update Services: Client-Server Protocol".

[MSDN-WindowsGDI] Microsoft Corporation, "Windows GDI", http://msdn.microsoft.com/en-us/library/dd145203.aspxx

[MSDN-XMLP] Microsoft Corporation, "A First Look at APIs For Creating XML Paper Specification Documents", http://msdn.microsoft.com/en-us/magazine/cc163664.aspx

[RFC1179] McLaughlin III, L., "Line Printer Daemon Protocol", RFC 1179, August 1990, http://www.ietf.org/rfc/rfc1179.txt

[RFC1831] Srinivasan, R., "RPC: Remote Procedure Call Protocol Specification Version 2", RFC 1831, August 1995, http://www.ietf.org/rfc/rfc1831.txt

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000, http://www.ietf.org/rfc/rfc2818.txt

[RFC2910] Herriot, R., Ed., Butler, S., and Moore, P., "Internet Printing Protocol/1.1: Encoding and Transport", RFC 2910, September 2000, http://www.ietf.org/rfc/rfc2910.txt

[RFC2911] Hastings, T., Ed., Herriot, R., deBry, R., et al., "Internet Printing Protocol/1.1: Model and Semantics", RFC 2911, September 2000, http://www.ietf.org/rfc/rfc2911.txt

[RFC4511] Sermersheim, J., "Lightweight Directory Access Protocol (LDAP): The Protocol", RFC 4511, June 2006, http://www.rfc-editor.org/rfc/rfc4511.txt

2   Functional Architecture

2.1   Overview

In addition to protocols supporting communication between print clients and print servers, the Print Services System also supports externally defined system protocols for the Group Policy System and the Active Directory System. The Print Services System uses the local file system services of the print server to store print jobs and printer drivers. The Print Services System also uses the local **registry** on the print server to persist the abstract data model (ADM) of the print server. The Print Services System has minimal interaction with other components of Windows.

Printers on a managed network are reflected in the Print Services System as print queues on a print server. Each print queue has an associated printer driver used by the Print Services System and applications to learn about printer capabilities, such as paper formats, color capabilities, and print quality. The associated printer driver is also responsible for conversion of application commands into the vendor-defined page description language (PDL) used to print a job on a printer.

Print clients submit print jobs to the print queues. The print queues are managed by a print spooler component running on the print server that buffers and orders print jobs arriving from many print clients simultaneously, or jobs arriving at a higher speed than the printers are capable of handling.

2.1.1   System Purpose

Printers are often acquired and deployed in an organization over time, resulting in a mixed infrastructure comprised of many different printers (with varying capabilities, such as color capabilities, paper sizes, resolutions, features, capacities, and 2D or 3D output). Some printers are intended for heavy use by large groups of users, and others for lighter use by smaller groups.

The Print Services System standardizes access and maintenance of multiple printers in a network, workgroup, or home environment so that administrators can more easily deploy and maintain printers, and so that users can choose the best one for a given job from among a pool of shared printers.

2.1.2   Functional Overview

2.1.2.1   Print Services System Components
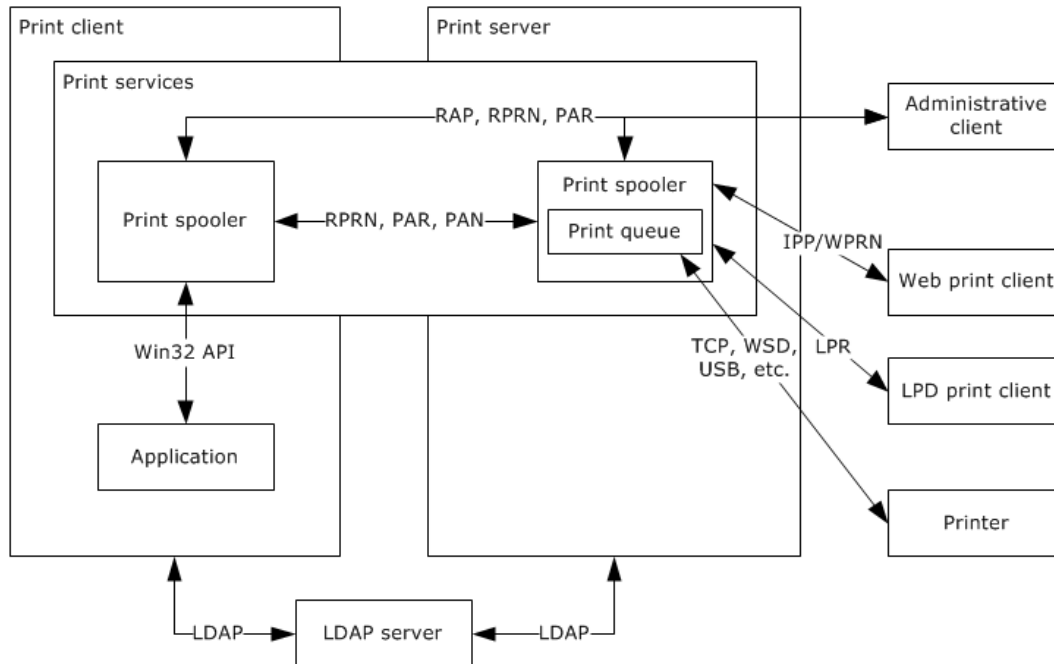
Figure 1: Components of printing services

2.1.2.2   Relationship of the Components within Print and Administrative Clients and Print Server

The diagrams in this section illustrate the relationship of the components within print (or administrative) clients, and the relationship of the components within print servers in the Print Services System. Additional details are available in System Internal Architecture, section 2.1.2.4.

2.1.2.2.1   Print and Administrative Clients
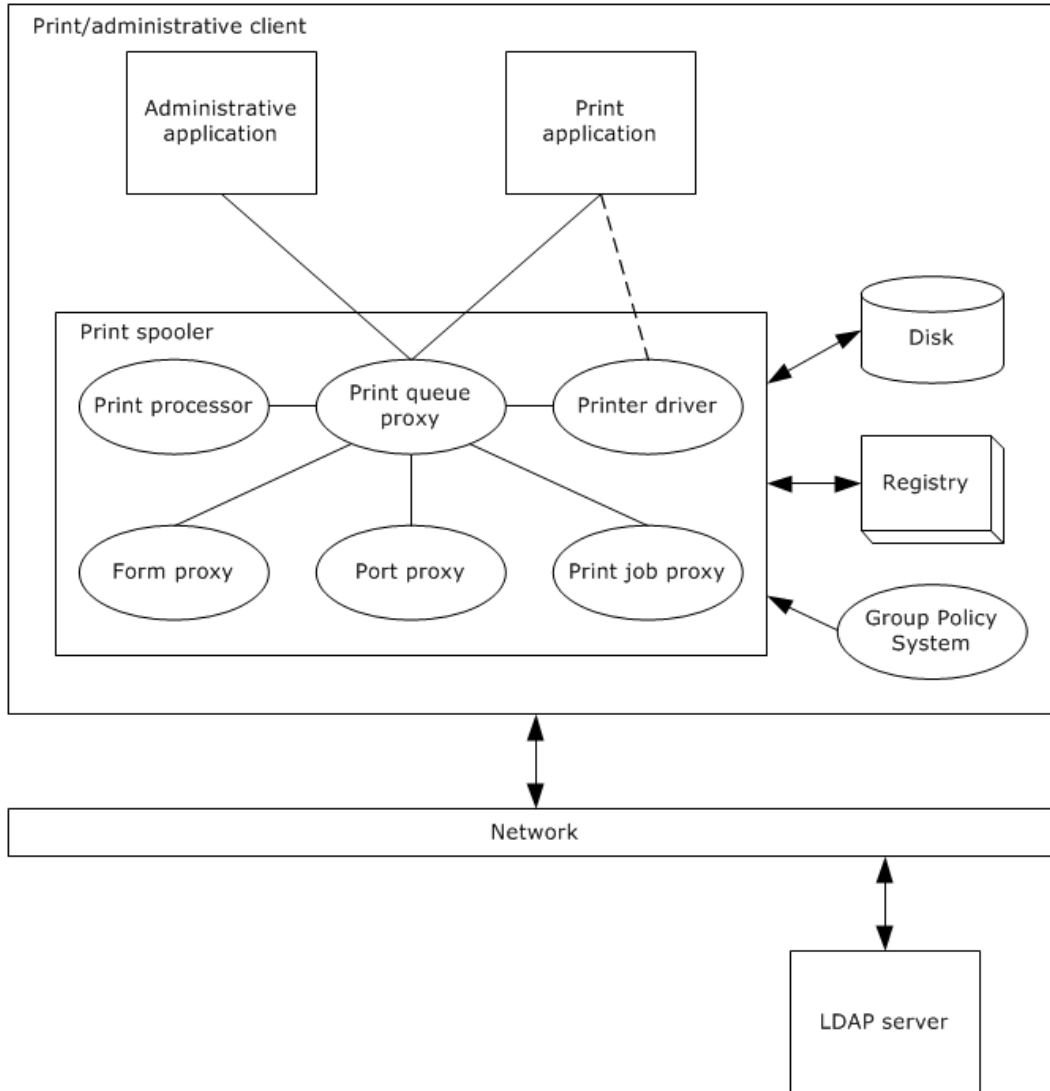
Figure 2: Components within print and administrative clients

2.1.2.2.2   Print Server
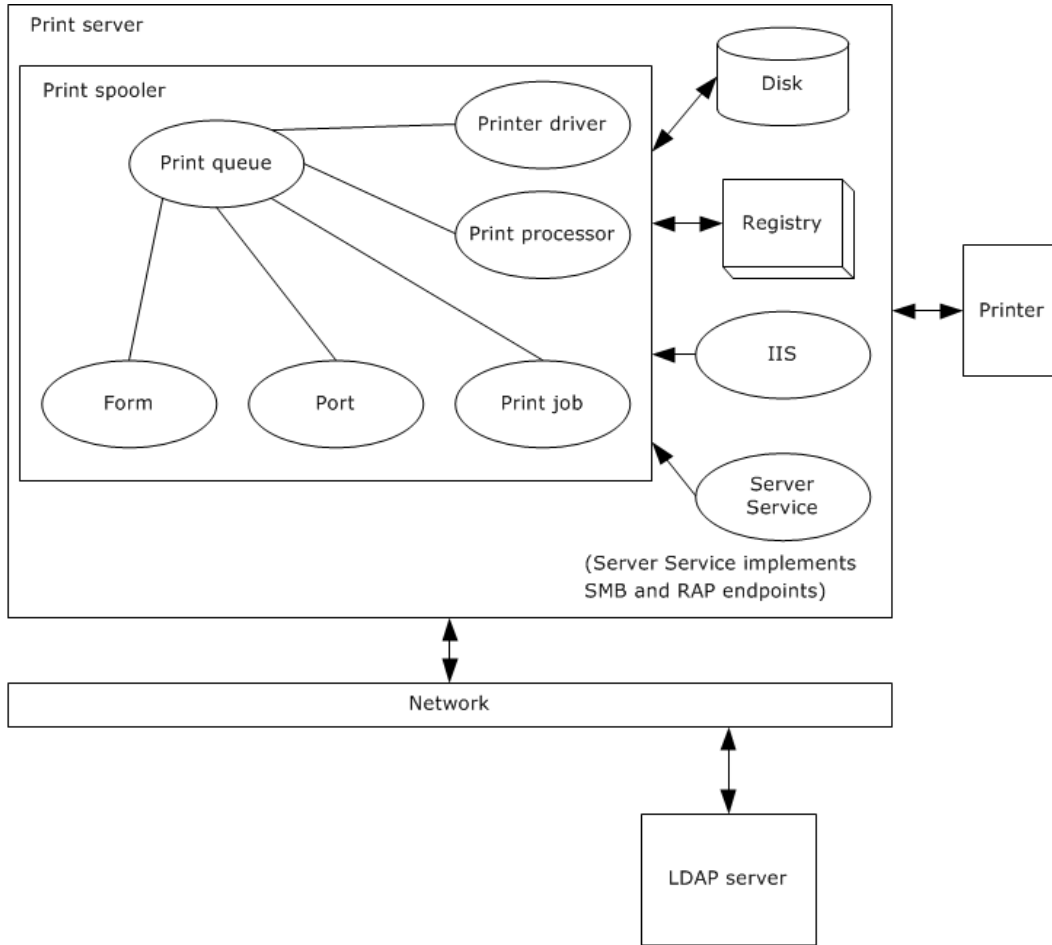
Figure 3: Components within the print server

2.1.2.3   Member Protocol Functional Relationships

The following diagrams illustrate the protocol relationships in the Print Services System.
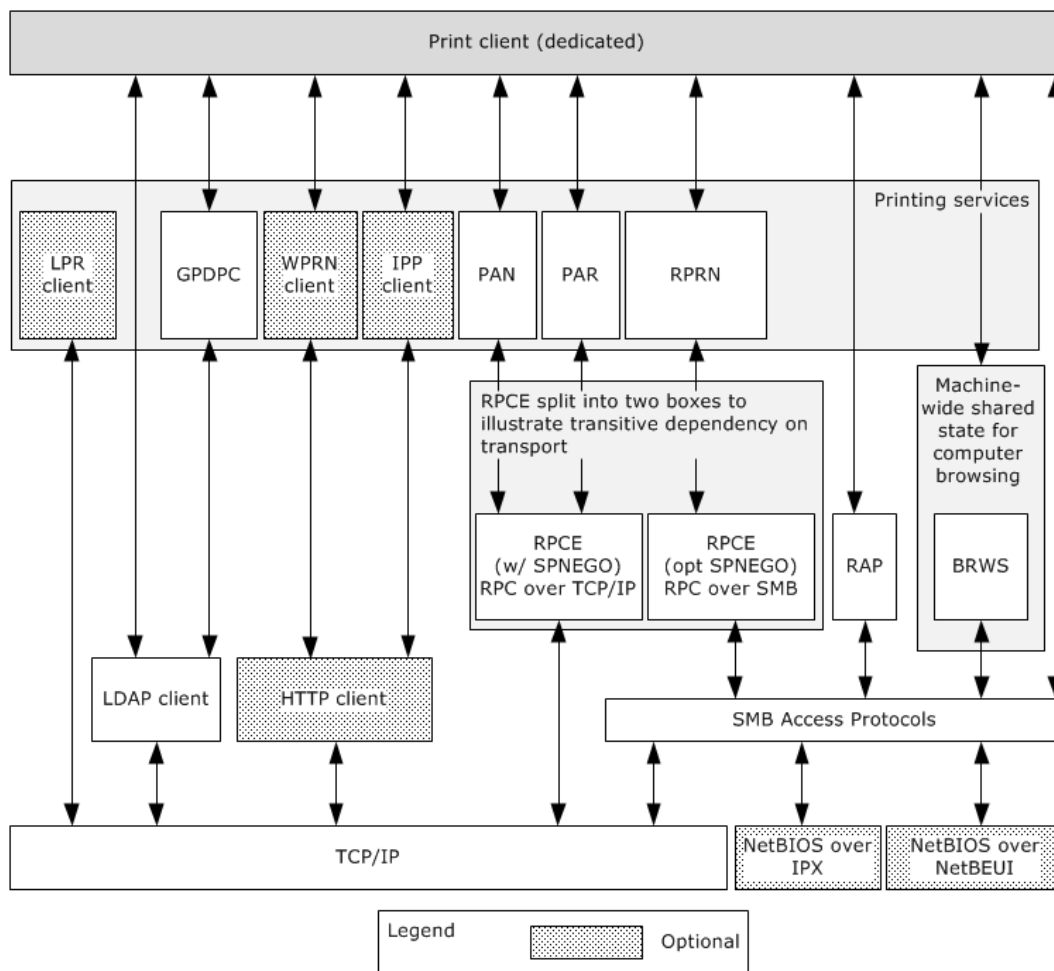
Figure 4: Full protocol layering for a print client depicting all protocols, including those that are optional
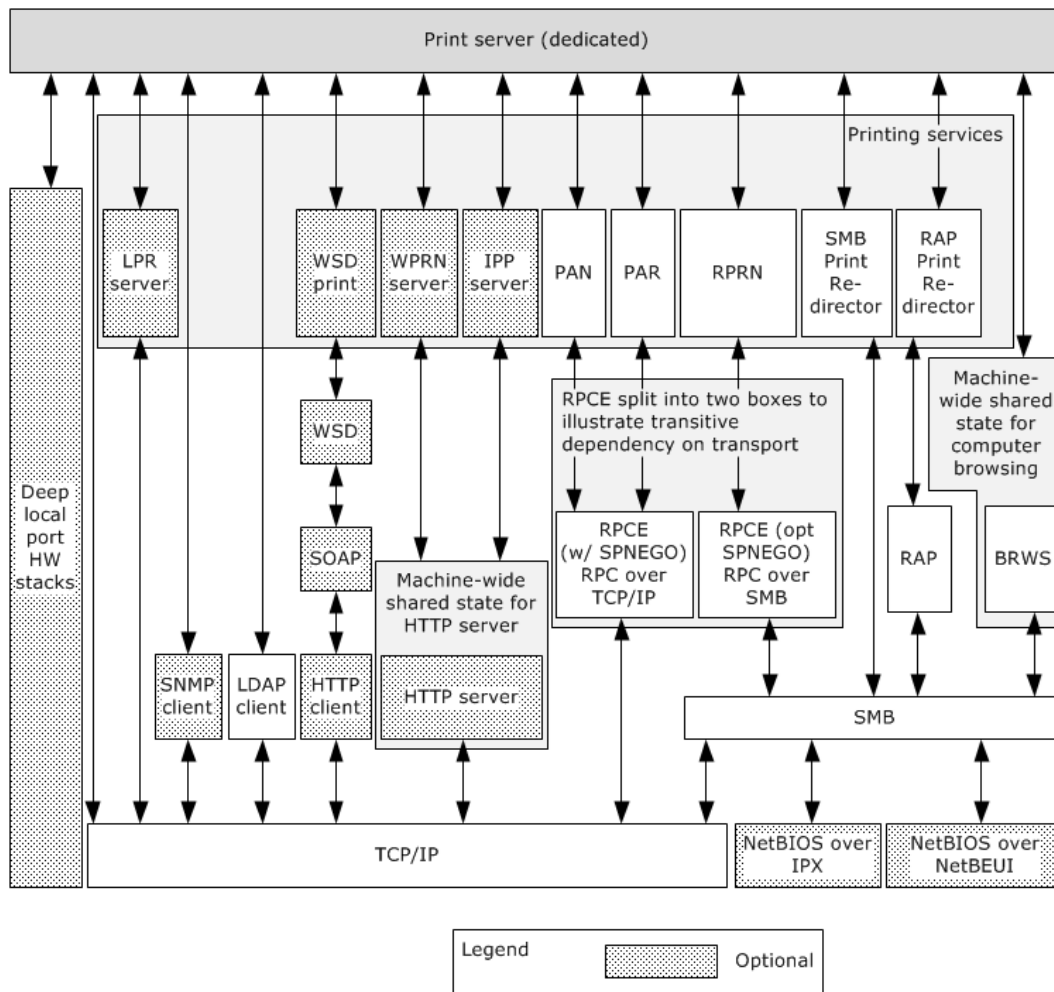
Figure 5: Full protocol layering for a print server depicting all protocols, including those that are optional
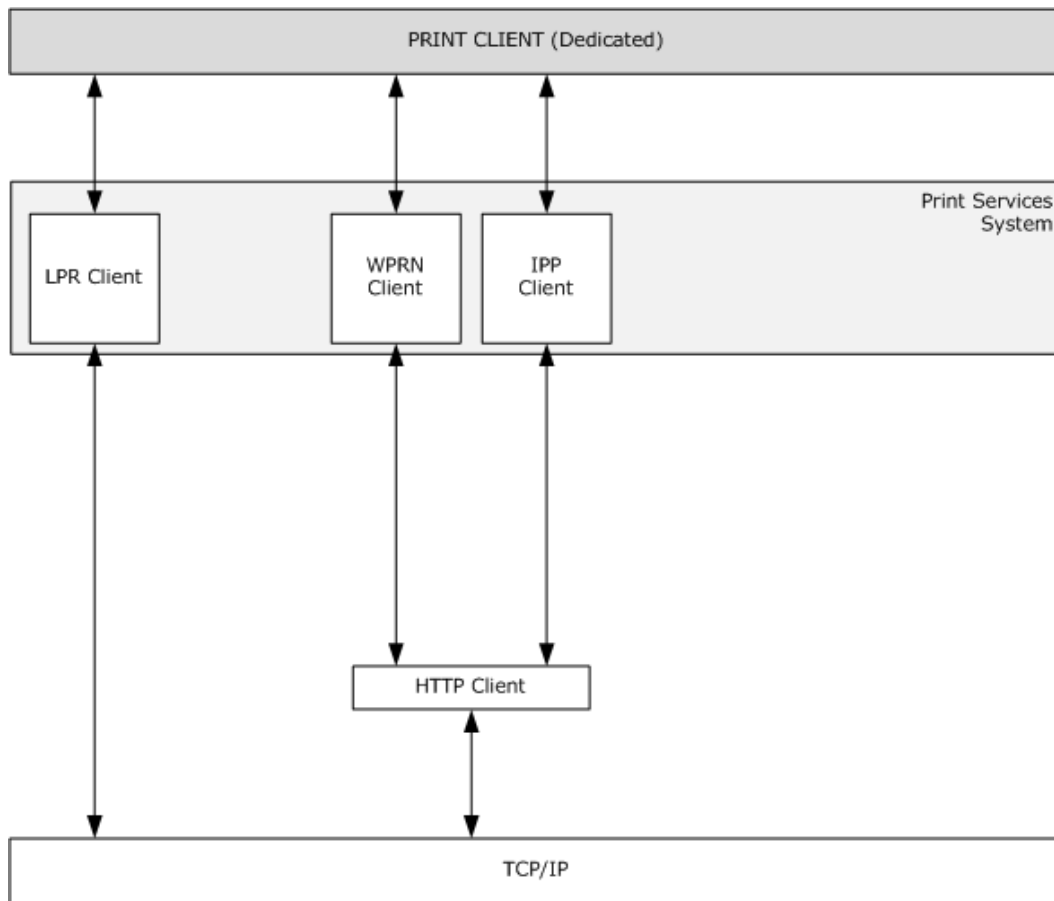
Figure 6: Optional protocol extensions for a print client
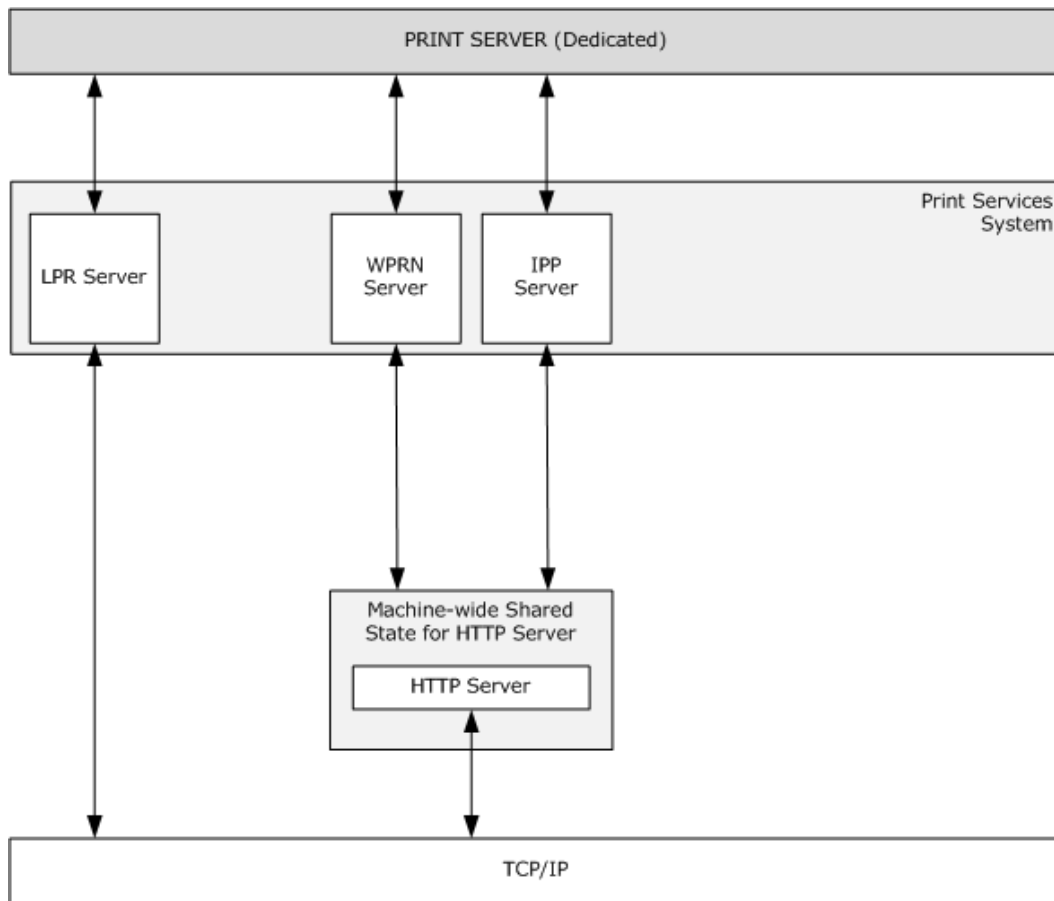
Figure 7: Optional protocol extensions for print server enabling a print server to support Internet and LPD print clients
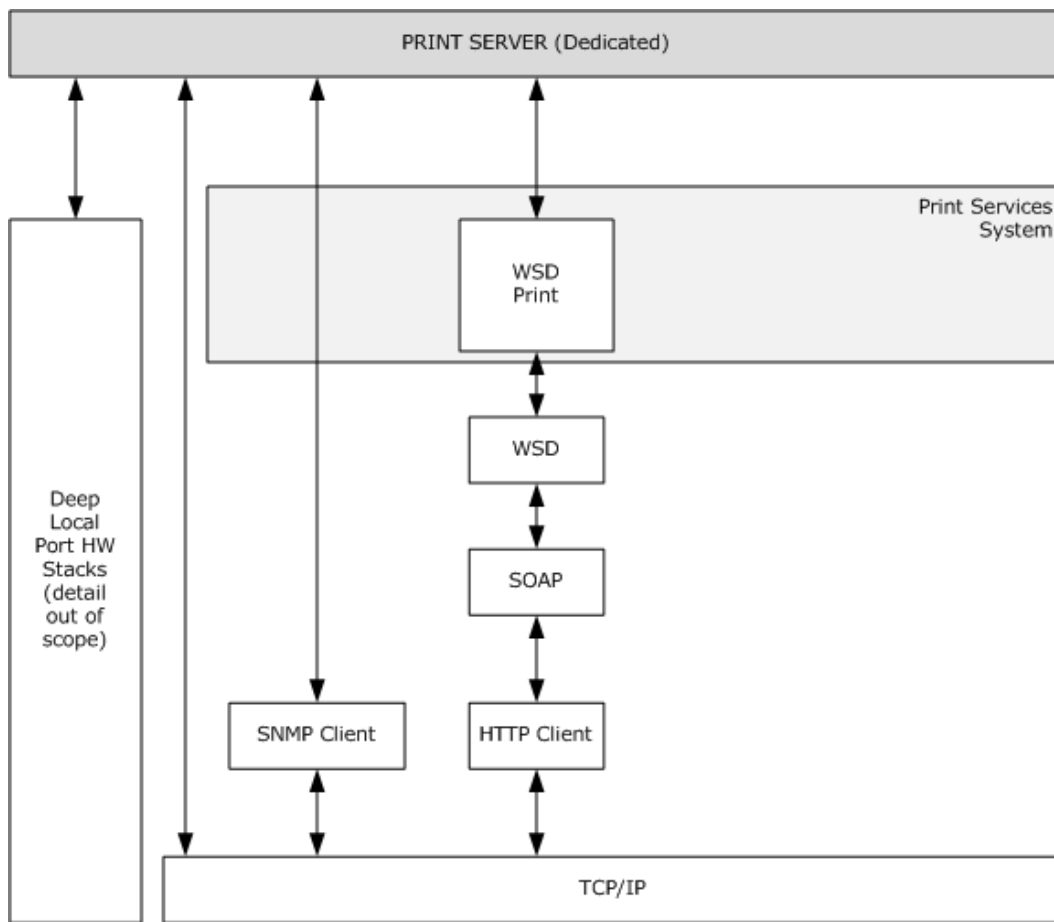
Figure 8: Protocol extensions used by print server to communicate with printers

2.1.2.4 System Internal Architecture
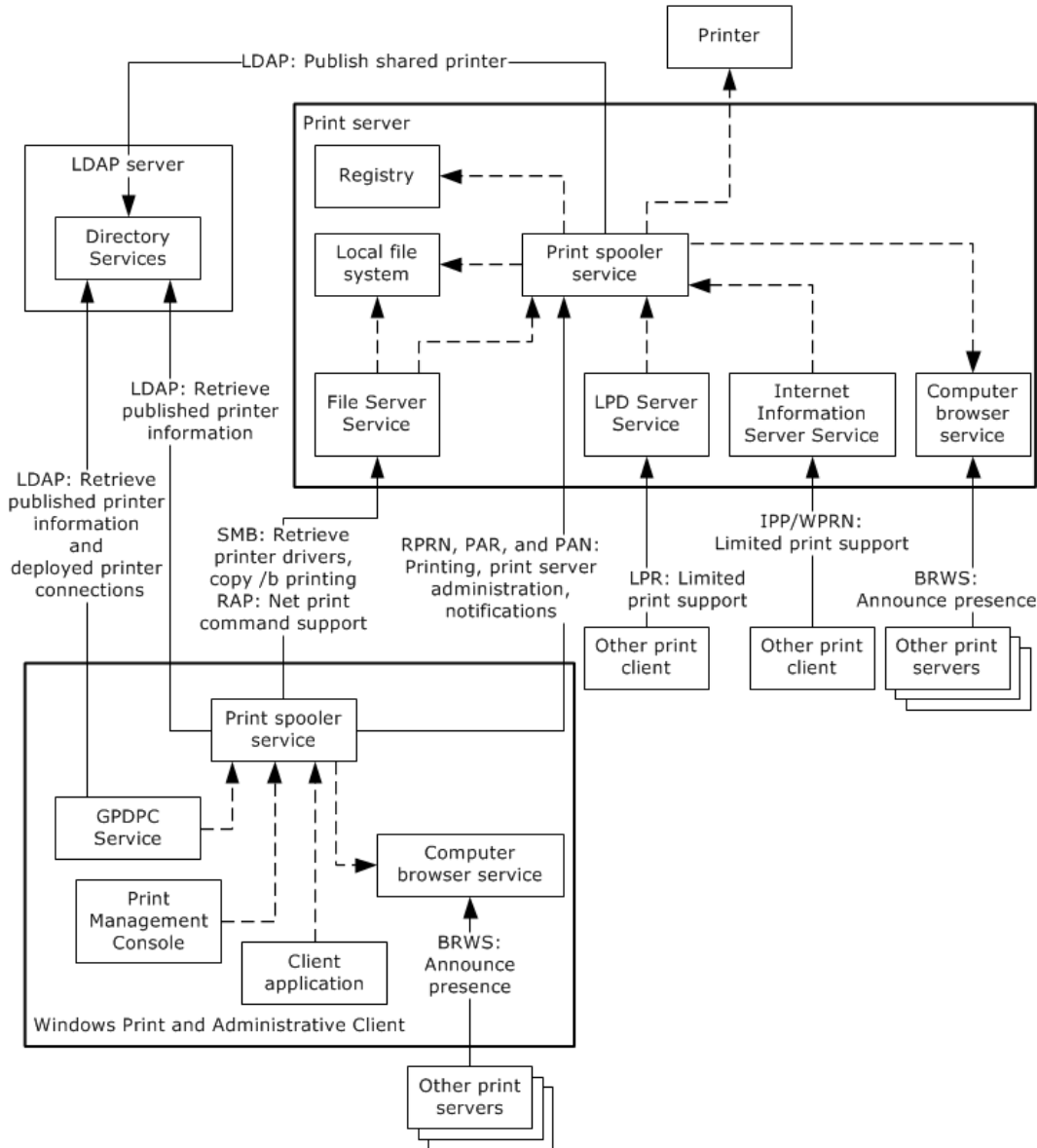
Figure 9: Print Services System internal system architecture

The Windows print client and Windows administrative client are shown in the preceding diagram as a single client object, contained by a bolded rectangle. Although they are often separate entities, these clients can share many of the same components and are differentiated by whether a client application or the Print Management Console (an administrative application under an administrator user account) is running.

Within the print client, the Group Policy Service component (GPDPC Service) retrieves Group Policy settings that restrict print client connections to specified print servers. The GPDPC Service also creates print queue connections per-computer and per-user.

To locate a print queue in a workgroup environment, the print spooler of the print client uses the local computer browser service component, which continually listens for print server announcement messages. In a domain environment, the print spooler of the print client can locate a print queue by various means; it can, for example, access an LDAP server of the Active Directory System. The print spooler then sends a print job to the print spooler installed on the print server.

The print server contains a print spooler that receives communications from the print client's print spooler by using the core protocols of the Print Services System (Print System Remote Protocol [MS-RPRN], Print System Asynchronous Remote Protocol [MS-PAR], and Print System Asynchronous Notification Protocol [MS-PAN]).

Three server service components provide print server support for print clients that do not use the core protocols:

- The File Server Service (using SMB/RAP) provides support for Microsoft Windows 98 operating system and Windows Millennium Edition operating system print clients, as well as users entering print commands at the command line of more current versions of Windows.

- The LPD Server Service (using the Line Printer Daemon Protocol [RFC1179]) provides support for UNIX/Linux print clients.

- The Internet Information Server Service (using the Internet Printing Protocol ([RFC2910] and [RFC2911]) and the Web Point-and-Print Protocol [MS-WPRN]) provides support for print clients when the print server is behind a firewall that prevents communications by way of the core protocols.

In a domain environment, the print server's print spooler publishes shared print queues (that correspond to shared printers) to an LDAP server of the Active Directory System. In a workgroup environment, the print server's print spooler finds other print servers by using the local Computer Browser Service, and uses the Print System Remote Protocol [MS-RPRN] and Print System Asynchronous Notification Protocol [MS-PAN] to announce shared print queues to each print server's local computer browser service.

The print server's print spooler persists system data in a registry component. The print spooler loads print driver, **port monitor**, print processor and **language monitor** modules by using the local file system. The print spooler also uses the local file system of the print server to buffer print jobs.

2.1.2.5   Windows Printing Architecture

Applications built by using Windows interfaces can use device-independent functions for creating print jobs and sending them to many types of printers. This device independence is based on the Windows architecture provided by the Windows graphics subsystems and the print spoolers and printer drivers. Print spoolers and printer drivers are replaceable and can be written to target specific client and printer hardware, while allowing applications to continue to use stable interfaces.

2.1.2.5.1   Print Client Communication with Print Server

Applications written by using the Windows Print API that are running on print clients access the local print spooler, which then redirects the application requests to the print services provided by the print server. This redirection establishes a level of indirection that isolates applications from protocol and capabilities negotiation and other complexities of the protocol, and keeps application code portable across multiple versions of the Print Services System.

The print spooler residing on the print server implements print queues. These print queues are associated with printer drivers, printer ports, and the printers to which they route print data. Print clients request print servers to perform a variety of functions, as shown in the following diagram.
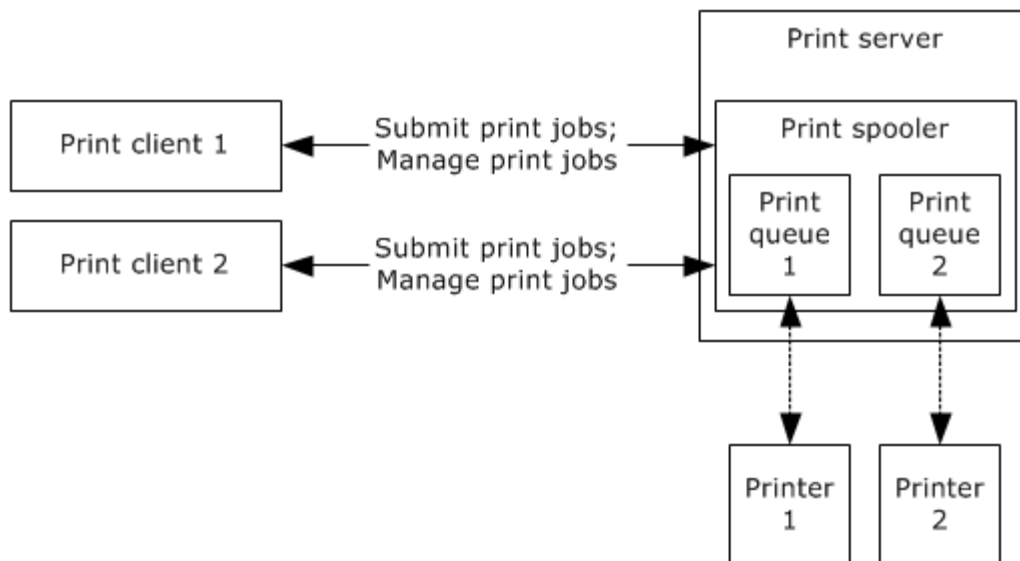
Figure 10: Print clients communicating with print server

2.1.2.5.2   Protocols Supporting Different Print Clients

Different Print Services System protocols and other protocols are used to allow a variety of print clients to communicate with a print server as shown in the following diagram. The Print Services System versions, capabilities, and protocol support are fully described in section 2.6.
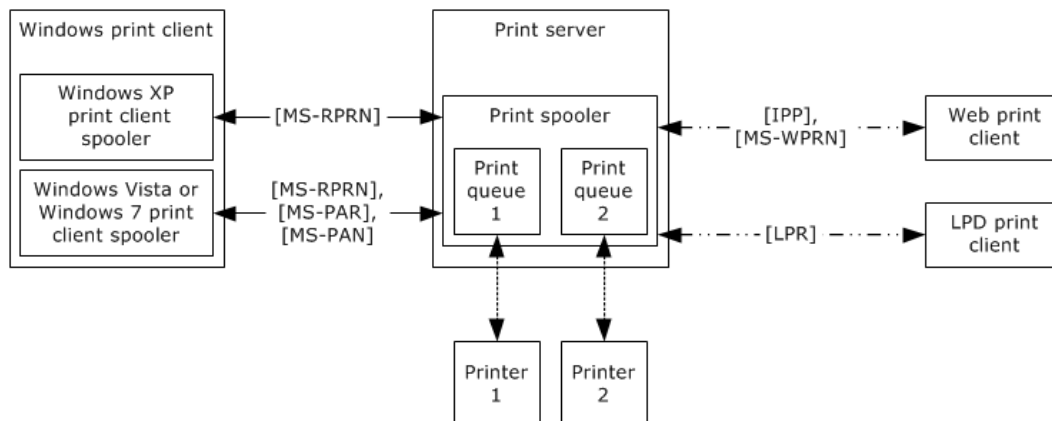
Figure 11: Protocols used by different print clients

2.1.2.5.3   Protocol Redirectors on Print Servers

Simultaneous server-side support for printing services and file access services by way of the SMB Access Protocols and the protocol described in [MS-RAP] require that redirector components be implemented on the server because each of these protocols exhibits print-specific and file access-specific functionality on a single protocol endpoint, and there is no abstract data model (ADM) common to all the print services and file access services.

These redirector components transparently route file access-specific requests to local calls to the file access services, and translate print-specific requests to local calls to the print spooler component

that is implementing the print server role. The redirector components then translate the results of these local calls back to the appropriate protocol response described in [MS-SMB], [MS-SMB2], or [MS-RAP].
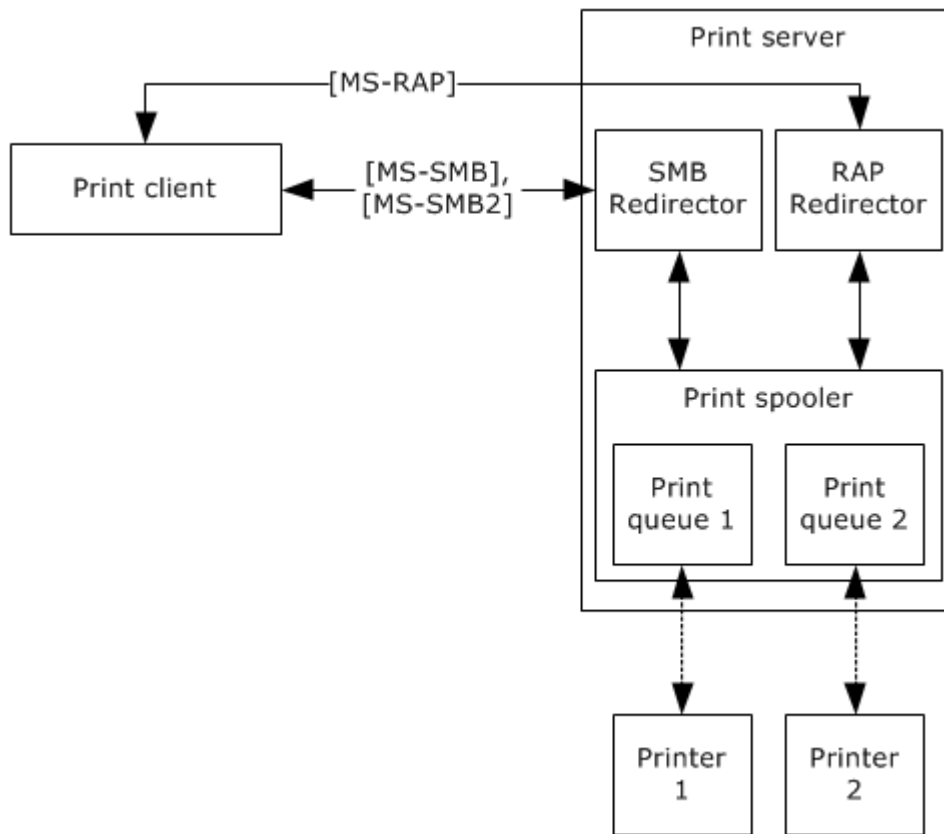
Figure 12: Redirectors routing data from a print client to the print spooler

2.1.2.5.4   Enabling Print Queues to Be Discoverable

To enable print queues to be discoverable by print clients, the Print Services System uses the Active Directory System, as described in [MS-ADOD], to store and provide information about shared queues. Although a print client can contact a print server directly to discover the print queues on the print server, using the Active Directory System implemented on the LDAP server allows a print client to enumerate print queues on multiple print servers. Furthermore, the Active Directory System isolates print clients from the requirement to locate print servers in the domain.

Print clients can request that print servers publish print queues to the Active Directory System through the Print System Remote Protocol and the Print System Asynchronous Remote Protocol. Print clients can then search for print queues in the Active Directory System using LDAP. The details of the interactions between the Print Services System and the Active Directory System are described in [MS-RPRN] section 2.3. Print clients search the Active Directory System using the mechanism described in [MS-RPRN] section 2.3.3.3.
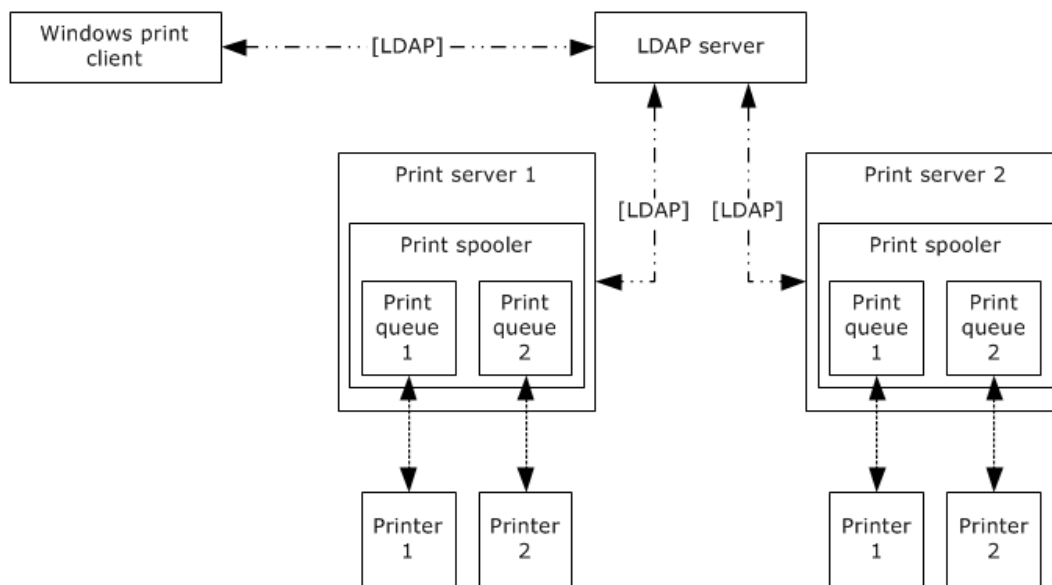
Figure 13: Obtaining information about print queues from an LDAP server

2.1.2.6 Translating Application Content to a Print Data Format

Windows works with the print spooler to manage the translation of application content to printer-specific data formats by using two configurations, as shown in the following sections. The print spooler determines the data format for the print job by using the mechanism described in [MS-RPRN] section 3.1.4.9.1.

In one configuration, Windows loads the printer driver directly into the application process. When the application prints, the data stream produced is already converted to the printer-specific data format in the application process, and is then passed to the print spooler service for buffering and further processing. This configuration is independent of the actual data format produced, and is referred to as producing a raw data format.

In another configuration, Windows does not load the printer driver into the application process, isolating the application from the actual translation work. Instead, Windows records document creation and drawing commands in a metafile, which is then passed to the print spooler service for buffering and conversion to a printer-specific data format. The metafile can have various data formats, such as the format described in [MS-EMFSPOOL] or the XML Paper Specification [MSDN-XMLP]. The print spooler service uses a print processor (usually part of the printer driver) to assist in the translation of the metafile to the printer-specific data stream. A print processor loads the metafile and replays the contained commands to the printer driver, which then performs the actual translation process to the printer-specific data stream.

2.1.2.7 Supporting Client-Side Rendering and Server-Side Rendering

Client-side rendering is one configuration of a client-server printing model. In this configuration, the print spooler service on the print client translates the metafile to the printer-specific raw format before sending it as a data-stream payload to the print spooler service on the print server.

In a server-side rendering configuration, the print spooler service on the print client sends an unmodified metafile payload to the print spooler service on the print server, where the metafile is subsequently converted to the printer-specific raw data format.

In order to support both client-side rendering and server-side rendering, it is a requirement that print processors and printer drivers are available to print servers and print clients. Windows addresses this requirement by using a mechanism called Point-and-Print, which allows a print client to download a printer driver directly from a print server, and a Package Point-and-Print mechanism that allows a print client to download a printer support package that includes the print processor (see [MS-WPRN] for more details).

2.1.2.8 Sending Print Data to a Printer via a Port Monitor

In order to send print data to a printer, a print server uses a port monitor module. Port monitors implement an abstraction API that allows the print spooler service to send and receive data through a printer port (such as USB or parallel ports) on the computer acting as the print server and connected to a printer, or through a network connection (such as TCP/IP) to a printer.

2.1.2.9 Branch Office Print Mode

An operating mode in which a print client is able to print directly to a print device instead of a print queue on a print server. It can reduce network costs in environments with centralized print servers. This mode is enabled by an administrator and is transparent to the user. Only print queues connected via the TCPMON monitor module (see [MS-RPRN] section 3.1.4.11.3) or WSDMON (Web Services for Devices Port Monitor) (see [MS-RPRN] section 3.1.4.11.4) are eligible to operate in **branch office print mode**.

2.1.3 Applicability

The Print Services System supports the use and management of a distributed print infrastructure. Using print servers and print clients that implement the member protocols described in this document, one or more printers can be shared between one or more print clients.

The Print Services System scales from workgroup use, in which printers are shared between computers; to domain-based networks, in which multiple print servers are employed in a cluster configuration, and the print client configuration is managed by the Active Directory System. The Print Services System also provides a subset of functionality for managing a single printer connected to a single computer.

In addition to protocols supporting communication between print clients and print servers, the Print Services System also supports externally defined system protocols for the Group Policy System and the Active Directory System. The Print Services System uses the local file system services of the print server to store print jobs and printer drivers. The Print Services System also uses the local registry on the print server to persist the ADM of the print server. The Print Services System has minimal interaction with other components of Windows.

Managed printers are reflected in the Print Services System as print queues on a print server. Each print queue has an associated printer driver used by the Print Services System and applications to learn about printer capabilities, such as paper formats, color capabilities, and print quality. The associated printer driver is also responsible for conversion of application commands into the vendor-defined page description language (PDL) used to print a job on a printer.

2.1.4 Relevant Standards

The Print Services System uses the following standards:

HTTP over TLS: Hypertext Transfer Protocol, as described in [RFC2818]. Used in the Print Services System as the transport protocol for the Internet Printing Protocol ([RFC2910] and [RFC2911]) and the Web Point-and-Print Protocol [MS-WPRN].

Internet Printing Protocol, as described in [RFC2910] and [RFC2911]. Used in the Print Services System for printing from Windows and non-Windows clients, or in cases where network security requirements mandate firewall settings that preclude use of the protocols described in [MS-PAR] or [MS-RPRN].

LDAP: Lightweight Directory Access Protocol, as described in [RFC4511]. Used by the print server role to publish shared printer information to directory, used by print client role to discover available shared printers in domain.

Line Printer Daemon Protocol, as described in [RFC1179]. Used in the Print Services System for printing from non-Windows clients, mainly UNIX and Linux systems (LPD clients).

RPC: Remote Procedure Call Protocol, as described in [RFC1831], [C706], and [MS-RPCE]. Used as transport for the protocols described in [MS-RPRN] (RPC over SMB), [MS-PAR] (RPC over TCP/IP), and [MS-PAN] (RPC over TCP/IP).

2.2 Protocol Summary

The member protocols of the Print Services System include the protocols described in this section, as well as the Internet Printing Protocol [RFC2911] [RFC2910] and the Line Printer Daemon Protocol [RFC1179]. The Print Services System uses the print data formats of the XML Paper Specification [MSDN-XMLP] and the enhanced metafile spool format [MS-EMFSPOOL].

The following table provides a comprehensive list of the member protocols of the Print Services System.

Protocol name	Description	Specification short name
Print System Remote Protocol	This protocol supports synchronous printing and spooling operations between a client and server, including print job control and Print Services System management. This protocol also provides status notifications (defined by the Print Services System) to the print client. An enhanced replacement for this protocol is described in [MS-PAR].	[MS-RPRN]
Print System Asynchronous Remote Protocol	This protocol supports printing and spooling operations between a client and server, including print job control and Print Services System management. This protocol also provides status notifications (defined by the Print Services System) to the print client. This protocol is designed to be used asynchronously by print clients whose implementations permit them to continue execution without waiting for a remote procedure call (RPC) method call to return. This protocol is an enhanced replacement for the Print System Remote Protocol [MS-RPRN].	[MS-PAR]
Print System Asynchronous Notification Protocol Specification	This protocol is used by print clients asynchronously to receive print status notifications from a print server and to send back responses to those notifications. A set of notifications and responses are defined together as a notification type. In contrast to the status notification capabilities included in the Print System Remote Protocol [MS-RPRN] and the Print System Asynchronous Remote Protocol [MS-PAR], the RPC interfaces and methods defined by this protocol provide a transport mechanism for arbitrary, IHV-extensible notification types. This protocol is used by IHV-provided components running on the print	[MS-PAN]

Protocol name	Description	Specification short name
	server to trigger the display of a user interface on the print client.	
Web Point-and-Print Protocol	This protocol is used in conjunction with the Internet Printing Protocol [RFC2911] [RFC2910] and enables a client to download printer driver software from a print server in the client network, from a website, or directly from a printer. Based on the Hypertext Transfer Protocol (HTTP).	[MS-WPRN]
Enhanced Metafile (EMF) Spool Format	The **EMFSPOOL** structure specifies a metafile format that can store a print job in portable form. The stored print job contains information for printing a document outside the control of the original application, either on the same computer or on another computer. An EMFSPOOL metafile is played back when its records are parsed and processed and the print job is sent to its destination.	[MS-EMFSPOOL]
SMB Access Protocols	Used in the Print Services System for communication with legacy Windows print clients and print servers (those that do not support the protocols described in [MS-RPRN] or [MS-PAR]) to submit print job information. This group of protocols is used by command-line-based copy to printer share operations.	[MS-CIFS], [MS-SMB], [MS-SMB2], and [MS-FSCC]
Remote Administration Protocol (RAP)	This protocol performs remote administrative functions, including share maintenance and printer maintenance on LAN Manager servers. It is used in the Print Services System for communication with legacy Windows print clients or print servers ("legacy" collectively used for those print clients and print servers that do not support the protocols described in [MS-RPRN] or [MS-PAR]) to manage print queues.	[MS-RAP]
Group Policy: Deployed Printer Connection Extension	This protocol supports managing connections to printers that are hosted by print servers and shared by multiple users. The print server component of this protocol enables an administrator to configure printer connections. The print client component of this protocol enables a user to discover the printer connections that have been configured.	[MS-GPDPC]

The protocols used by the Print Services System perform roles as follows:

Printing: The protocols described in [MS-RPRN], [MS-PAR], [RFC2911], [RFC2910], [RFC1179], and the SMB Access Protocols are used for this role.

Managing print jobs: The protocols described in [MS-RPRN] and [MS-PAR] are used for this role. The protocol described in [MS-RAP] is used for this role for print clients and print servers that do not support the protocols described in [MS-RPRN] and [MS-PAR].

Managing the Print Services System: The protocols described in [MS-RPRN] and [MS-PAR] are used for this role. The protocol described in [MS-RAP] is used for this role for print clients and print servers that do not support the protocols described in [MS-RPRN] and [MS-PAR].

Receive notifications about general printing status: The protocols described in [MS-PAR], [MS-RPRN], and [MS-PAN] are used for this role.

Receive notifications about specific printing status from IHV-defined components: The protocol described in [MS-PAN] is used for this role.

Respond to notifications about specific printing status to IHV-defined components: The protocol described in [MS-PAN] is used for this role.

Download printer drivers to a client: The protocols described in the SMB Access Protocols and [MS-WPRN] (based on HTTP) are used to download printer drivers from a website, a printer, or a print server in the same network as the print client.

Store print jobs: The protocols described in [MS-EMFSPOOL] and the XML Paper Specification [MSDN-XMLP] are used as payloads for print jobs.

Configure print clients: The protocol described in [MS-GPDPC] is used for this role.

The following tables shows the member protocols of the Print Services System grouped according to their primary purpose.

Protocols in this table enable print queue connection and printing from internet clients.

Protocol name	Description	Specification short name
Internet Printing Protocol	Used in the Print Services System for printing from non-Windows clients, or in cases where network security requirements mandate firewall settings that preclude use of the protocols described in [MS-PAR] or [MS-RPRN].	[RFC2911], [RFC2910]
Web Point-and-Print Protocol	This protocol is used in conjunction with the Internet Printing Protocol and enables a client to download printer driver software from a print server in the client network, from a website, or directly from a printer. Based on the Hypertext Transfer Protocol (HTTP).	[MS-WPRN]

Protocols in this table allow printing and IHV plug in-defined communication.

Protocol name	Description	Specification short name
Print System Asynchronous Remote Protocol	This protocol supports printing and spooling operations between a client and server, including print job control and Print Services System management. This protocol also provides status notifications (defined by the Print Services System) to the print client. This protocol is designed to be used asynchronously by print clients whose implementations permit them to continue execution without waiting for a remote procedure call (RPC) method call to return. This protocol is an enhanced replacement for the Print System Remote Protocol [MS-RPRN].	[MS-PAR]
Print System Remote Protocol	This protocol supports synchronous printing and spooling operations between a client and server, including print job control and Print Services System management. This protocol also provides status notifications (defined by the Print Services System) to the print client. An enhanced replacement for this protocol is described in [MS-PAR].	[MS-RPRN]
Print Asynchronous Notification	This protocol is used by print clients asynchronously to receive print status notifications from a print server and to send back responses to those notifications. A set of notifications and	[MS-PAN]

Protocol name	Description	Specification short name
Protocol	responses are defined together as a notification type. In contrast to the status notification capabilities included in the Print System Remote Protocol [MS-RPRN] and the Print System Asynchronous Remote Protocol [MS-PAR], the RPC interfaces and methods defined by this protocol provide a transport mechanism for arbitrary, IHV-extensible notification types. This protocol is used by IHV-provided components running on the print server to trigger the display of a user interface on the print client.	

Protocols in this table enable printing from clients only capable of Print Services System (PSS) version 1.0 support (see section 2.6 regarding PSS versioning).

Protocol name	Description	Specification short name
Remote Administration Protocol (RAP)	This protocol performs remote administrative functions, including share maintenance and printer maintenance on LAN Manager servers. It is used in the Print Services System for communication with legacy Windows print clients or print servers ("legacy" collectively used for those print clients and print servers that do not support the protocols described in [MS-RPRN] and [MS-PAR]) to manage print queues.	[MS-RAP]
SMB Access Protocols	Used in the Print Services System for communication with legacy Windows print clients and print servers (those that do not support the protocols described in [MS-RPRN] and [MS-PAR]) to submit print job information. This group of protocols is used by command-line-based copy to printer share operations.	[MS-CIFS], [MS-SMB], [MS-SMB2], and [MS-FSCC]

2.3 Environment

The following sections identify the context in which the system exists. This includes the systems that use the interfaces provided by this system of protocols, other systems that depend on this system, and, as appropriate, how components of the system communicate.

2.3.1 Dependencies on This System

None.

2.3.2 Dependencies on Other Systems/Components

The Print Services System member protocols require the following protocols:

- SMB Access Protocols:

  - Used as transport for the protocol defined in [MS-RPRN] (uses RPC over named SMB pipes).

  - Used for driver download from print server (using the copy file functionality of the SMB Access Protocols).

- RAP protocol [MS-RAP], used to browse for printers.

- CIFS protocol [MS-CIFS], used to copy drivers from the server and to submit print jobs over redirected ports.

- SMB Version 1.0 protocol [MS-SMB] plus RAP protocol [MS-RAP] (SMB Version 2.0 protocol [MS-SMB2], however, is not compatible with RAP):

    - Used for managing SMB print scenarios, using the protocol defined in [MS-RAP].

    - Used for managing SMB file copy functionality of the print server, using the protocol defined in [MS-RAP].

- HTTP/HTTPS:

    - Used as transport for the protocol defined in [MS-WPRN].

    - Used as transport for the Internet Printing Protocol ([RFC2910] and [RFC2911]).

- RPC over TCP/IP:

    - Used as transport for the protocol defined in [MS-PAR].

    - Used as transport for the protocol defined in [MS-PAN].

- Group Policy: Core Protocol [MS-GPOL] (requires prerequisites [MS-NRPC] and [MS-DRSR]).

    - Used for the protocol defined in [MS-GPDPC].

- LDAP

    - Used for publishing and query of shared print queues.

    - Used by the protocol defined in [MS-GPDPC].

- TCP/IP

    - Used as transport by LDAP.

    - Used as transport by the Line Printer Daemon Protocol [RFC1179].

    - Used as transport by HTTP.

    - Used as transport by HTTPS.

    - Used as transport by RPC over TCP/IP.

    - Used as transport by the SMB protocol family (unless on an IPX or NetBEUI network).

- Common Internet File System (CIFS) Browser Protocol Specification [MS-BRWS].

    - Used for workgroup preconditions.

The Print Services System member protocols, as defined in [MS-RPRN], [MS-PAR] and [MS-PAN], are RPC-based and require RPC bindings between print servers and print clients to registered RPC endpoints. If a print client or print server is unable to register an RPC endpoint or create RPC bindings, then Print Services provided by the local print spooler will only allow the management and use of locally connected printers. Firewalls implemented on print clients or print servers are configured so that all ports required by member protocols are open for RPC-based communication, or at least open for HTTP-based communication to support the internet printing scenario using the Internet Printing Protocol ([RFC2910] and [RFC2911]) and the protocol described in [MS-WPRN].

The Print Services System can run in a domain-based network and workgroup environment. Print Services require that File Share services are installed and enabled (for Point and Print driver download and for SMB and Remote Administration Protocol [MS-RAP] support). Upon creation of a printer queue connection to a print server, the print client tries to copy the driver from the print server (see Use Cases in sections 2.5.3.3 and 2.5.3.4). If that fails, the print client tries, in turn:

- Any available implementation-specific driver sources or repositories<1>.

- The print client prompts the user to supply an appropriate printer driver from a disk, an Internet location, or other media.

If the print client cannot obtain a suitable printer driver from any of these sources, the print queue connection cannot be created.

The Print Services System requires the protocol defined in [MS-BRWS] for workgroups. Availability of this protocol and the Active Directory System is not determined at startup, but only upon request. If neither the Common Internet File System (CIFS) Browser Protocol [MS-BRWS] nor the Active Directory System are available, a print client will not list any available shared print queues in the network, and as a result, only local functionality of the print spooler is available, unless the user of the print client knows the name of the print server and enters it manually.

The Print Services System makes use of the Windows Update service, as described in [MS-WUSP], if available. Windows Update is used to find the most up-to-date or best matching printer drivers for a print queue connection. If Windows Update services are unavailable, print services use other methods to find an appropriate printer driver (see Use Cases in sections 2.5.3.2 and 2.5.3.3).

All member protocols can be used between print clients and print servers, depending on circumstances.

The Print Services System uses the Active Directory System in domain-based networks. The LDAP protocol [RFC4511] is used between print server and the Active Directory System and between print client and the Active Directory System. In workgroup environments, the Common Internet File System (CIFS) Browser Protocol [MS-BRWS] is used for communication between print clients and print servers, and between print servers.

The file system access used by clients (to download printer drivers only) assumes a directory structured file system, and the print server generally restricts access using **access control lists (ACLs)**, so that print clients have only read-access to the printer driver files. The file system supports file date/time stamps for driver version comparison purposes.

Any number of print servers and print clients can be operated in a domain-based network. The number of shared print queues clients retain knowledge of in a Windows workgroup is limited to an implementation-defined maximum,<2> which limits the useful number of shared printers in a Windows workgroup network. From the system perspective, the entirety of print clients and print servers in a given network represents a single instance of the system.

The Print Services System is also dependent on the following components and infrastructure.

- Windows Authentication Services System, as described in [MS-AUTHSOD].

- Active Directory System (in a domain environment): If the Print Services System is deployed within a domain, print servers can publish shared printers to the Active Directory System using LDAP. Print clients joined to domains also require access to the Active Directory System to discover published printers.

- Networking system to connect print server, print clients, and the Active Directory System.

- Printers.

- Application software that provides printing functionality, such as Microsoft Office Word.

The Print Services System is influenced by the following:

- The Active Directory System can propagate policy settings controlling local spooler behavior to print clients and print servers<3> using the protocol defined in [MS-GPOL].

- These policies control print spooler implementation aspects, for example which print servers are considered trusted for printer driver download in Point-and-Print scenarios.

- The protocol described in [MS-GPDPC] influences the Print Services System in regards to **deployed printer connections**.

2.4   Assumptions and Preconditions

The following assumptions and preconditions are necessary for the Print Services System to operate successfully.

- A network is available to provide a viable transport for communication between the server and its clients.

- The transport protocol for that network is available and configured (for example, the TCP transport is configured with a valid IP address).

- **Security providers** are available to the system to provide message authentication and security.

- The durable storage devices used to store the system's state is on all participating computers.

- The print spooler service is in the Running state on the server and each of the clients participating in the interaction.

- The print spooler is installed on all the computers involved.

- The local client connecting to a remote print server must support impersonation if the remote spooler requires authentication.

- The remote print spooler is not required to use impersonation, nor does it require that local clients support impersonation, except when using the protocol defined in [MS-PAN].

- The file (SMB Protocol Family support) server service is in the running state on the server.

- The print client and print server support the protocol defined in [MS-RPRN].

- The system is configured so that participants can access its services locally or remotely.

- It is assumed that each participant is trusted by the system.

- If Member Protocols supported by the system, as listed in section 2.2, have additional assumptions and preconditions for when that protocol is in use, please see the relevant member protocol specification for details.

- Print servers have a durable store to place the following objects in:

  - Printer drivers

  - Print jobs

- Print clients have a durable store to place the following objects in:

  - Printer drivers

- Print clients and print servers have access to local storage to persist ADM and state information; under Windows, this local storage is in the registry.

- In a domain configuration, print clients and print servers have access to the Active Directory System provided by the domain.

- **Authentication** services supporting Simple and Protected Negotiate (SPNEGO, as described in [MS-SPNG]) are available to the print servers and clients.

- The Print System Remote Protocol, as described in [MS-RPRN], is a remote procedure call (RPC) interface and therefore has the prerequisites described in [MS-RPCE] section 1.5 as being common to RPC interfaces.

- The Print System Asynchronous Remote Protocol, as described in [MS-PAR], is a RPC interface and therefore has the prerequisites described in [MS-RPCE] section 1.5 as being common to RPC interfaces.

- The Print Asynchronous Notification Protocol, as described in [MS-PAN], is a RPC interface and therefore has the prerequisites described in [MS-RPCE] section 1.5 as being common to RPC interfaces.

- The print client has obtained the name of the print server that supports the Print Services System. Various protocols can be used to accomplish this, such as [MS-ADLS], [MS-ADSC], and [MS-BRWS].

2.5 Use Cases

The Print Services System is designed to support scenarios that allow users shared access to printers connected to print servers or to computers used by others. When a user wants to print to a shared printer, the system facilitates this action by providing the following:

- Lists of shared printers

- Connecting the user to the print queue associated with the desired printer and enabling the user to install the appropriate printer driver

- Transporting the print data from the print client to the print queue

- Processing multiple print jobs in order of their arrival or priorities

- Keeping the user apprised of the status of the print job

The Print Services System also facilitates the management of a large number of shared printers across multiple print servers by enabling an administrator to remotely install and configure shared print queues on print servers, to manage the use of shared print queues on print servers, and to view and manage print jobs that have been submitted by all users to shared print queues on print servers.

Although there is a distinction between the roles and capabilities of administrator and user when participating in the Print Services System, both roles can be conducted on the same computer. And although there is a distinction between a print client and a print server as actors in scenarios that are using the Print Services System, a print client can assume the role of a print server if it enables

other print clients to discover and use (or administrative clients to administer) a print queue installed on the print client computer.

This section provides the use cases that describe the functionality of the Print Services System in terms of actors that participate in this system and their goals. The use case participants most often include a print client and a print server, and they are triggered by a user performing print operations, or an administrator performing administrative operations.

Section 2.5.2 provides a table that summarizes the use cases described in section 2.5.3.

2.5.1 Actors

Print client: The print client connects to a print queue on a print server, sends print data to the print queue, displays notifications about the printer or print job, and disconnects from the print queue. The print client can take the form of an application that the user is printing from, or as a component of an operating system.

Administrative Client: The administrative client connects to print servers to establish, configure, and manage print queues hosted on the print servers. The administrative client can also be used to manage print jobs sent to the print queues. The administrative client takes the form of an administrative tool application.

Print Server: The print server hosts print queues, each print queue corresponding to a shared printer. The print server stores and uses printer drivers that are necessary for processing print data before it is sent to printers. Depending on the configuration of the print server and print client, the print server can supply printer drivers to print clients, so print clients can perform print data processing locally. The print server processes requests from print clients and the administrative clients (such as connect, disconnect, upload driver, download driver, route print data). The print server manages access from users and administrators to the print queues, sends notifications about print queues and print jobs to users and administrators, and routes print data to the appropriate print queues.

Internet Browser: An Internet browser can be used to discover shared print queues on a print server. The internet server component on the print server uses local print spooler APIs to obtain a list of print queues and presents the list to the client as an HTML page. An Internet browser can be used for this purpose if a firewall or other network restrictions do not allow the core Print Services System protocols to be used. When a user knows the name of the print queue, the user can enter the address in the print client user interface to connect to the printer.

User: The individual who uses a print client to choose a print queue from a list of available shared print queues, sends a print job to the print queue, and monitors the progress of the print job.

Administrator: The individual responsible for installing and configuring printers within a deployed Print Services System, as well as monitoring, managing, and troubleshooting printing functions. The administrator uses an administrative client.

Active Directory System: The Active Directory System is used to store a list of shared print queues in a domain environment. The Print Services System uses version 3 of the LDAP protocol [RFC4511] to publish information about shared print queues to the Active Directory System. Print clients can then discover shared print queues that are published by the Active Directory System, allowing a user to choose a print queue to which to send a print job.

Group Policy System: The Print Services System makes use of the Group Policy System, as described in [MS-GPOD], to restrict print clients from accessing specified print servers, as well as to remotely push pre-configured print queue connections to print clients. The Print Services System

uses a Group Policy Services extension protocol, as described in [MS-GPDPC], to distribute these pre-configured print queue connections to print clients.

2.5.2 Use Case Summary Diagrams

The following table provides an overview of use cases which span the functionality of the Print Services System. Use case extensions are noted within each use case. Detailed descriptions for these use cases are provided in section 2.5.3.

Use case group	Use cases
Provision or Delete a print queue	Provision a print queue using the protocol described in [MS-RPRN] - Administrative Client ▪ Install print processor on print server Provision a print queue using the protocol described in [MS-PAR] - Administrative Client ▪ Install print processor on print server Delete a print queue using the protocol described in [MS-RPRN] - Administrative Client ▪ Delete objects on print server that the implementation determines are no longer needed. Delete a print queue using the protocol described in [MS-PAR] - Administrative Client ▪ Delete objects no longer referenced on print server
Locate and Establish a Connection to a print queue	Locate and establish a connection to a print queue in a domain environment using the protocol described in [MS-RPRN] - print client 1. Print client downloads printer driver 2. Print client connections are restricted by Group Policy settings 3. Print client registers for notifications of print queue status Locate and establish a connection to a print queue in a domain environment using the protocol described in [MS-PAR] - print client 1. Print client downloads printer driver 2. Print client connections are restricted by Group Policy settings 3. Print client registers for notifications of print queue status Locate and establish a connection to a print queue in a workgroup environment using the protocol described in [MS-RPRN] - print client 1. Print client downloads printer driver 2. Print client registers for notifications of print queue status Locate and establish a connection to a print queue in a workgroup environment

Release: Thursday, February 13, 2014

Use case group	Use cases
	using the protocol described in [MS-PAR] - print client
	1. Print client downloads printer driver
	2. Print client registers for notifications of print queue status
	Locate and establish a connection to a print queue from an internet client using the Internet Printing Protocol ([RFC2911] and [RFC2910]); download a printer driver using the protocol described in [MS-WPRN] - print client
Manage a print queue	Set Permissions on a print queue using the protocol described in [MS-RPRN] - Administrative Client
	Set Permissions on a print queue using the protocol described in [MS-PAR] - Administrative Client
Submit a Print Job	Submit a print job using the protocol described in [MS-RPRN] - print client
	1. Print client obtains notifications about print job status
	2. Print client obtains notifications for IHV-defined components on print server
	Submit a print job using the protocol described in [MS-PAR] - print client
	1. Print client obtains notifications about print job status
	2. Print client obtains notifications for IHV-defined components on print server
	Submit a print job using the Internet Printing Protocol ([RFC2911] and [RFC2910]) - print client
	Submit a print job using SMB Protocol Family - print client
Manage a Print Job	Manage print jobs submitted by self using the protocol described in [MS-RPRN] - print client
	Manage Print jobs submitted by self using the protocol described in [MS-PAR] - print client
	Manage print jobs submitted by All users using the protocol described in [MS-RPRN] - administrative client
	Manage print jobs submitted by All users using the protocol described in [MS-PAR] - administrative client
	Manage a print job submitted from command line using the protocol described in [MS-RAP] - print client

The following use case diagrams illustrate the use cases described in this section, dividing them between those initiated by an administrative client and those initiated by a print client.
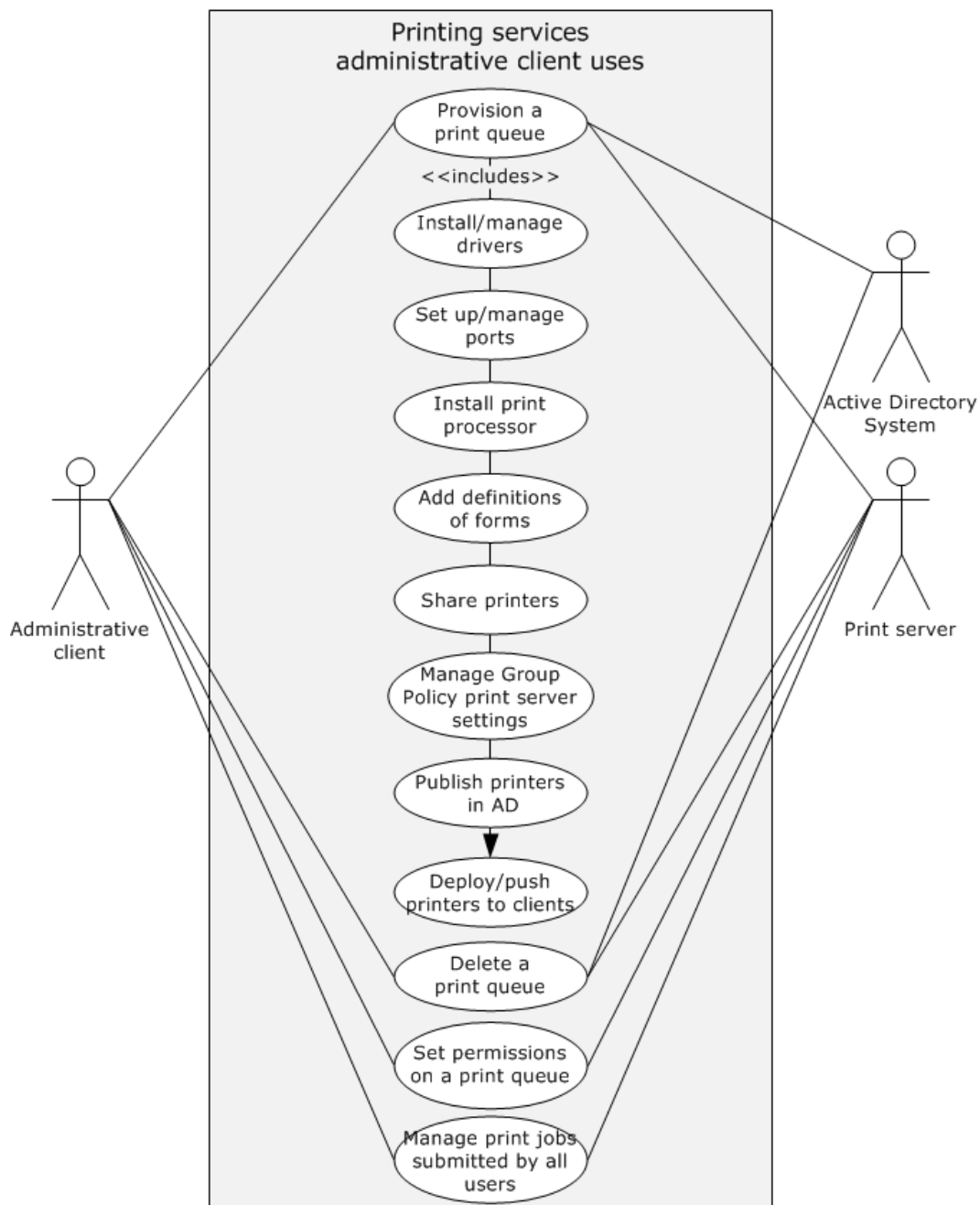
**Figure 14: Printing services use cases initiated by an administrative client**
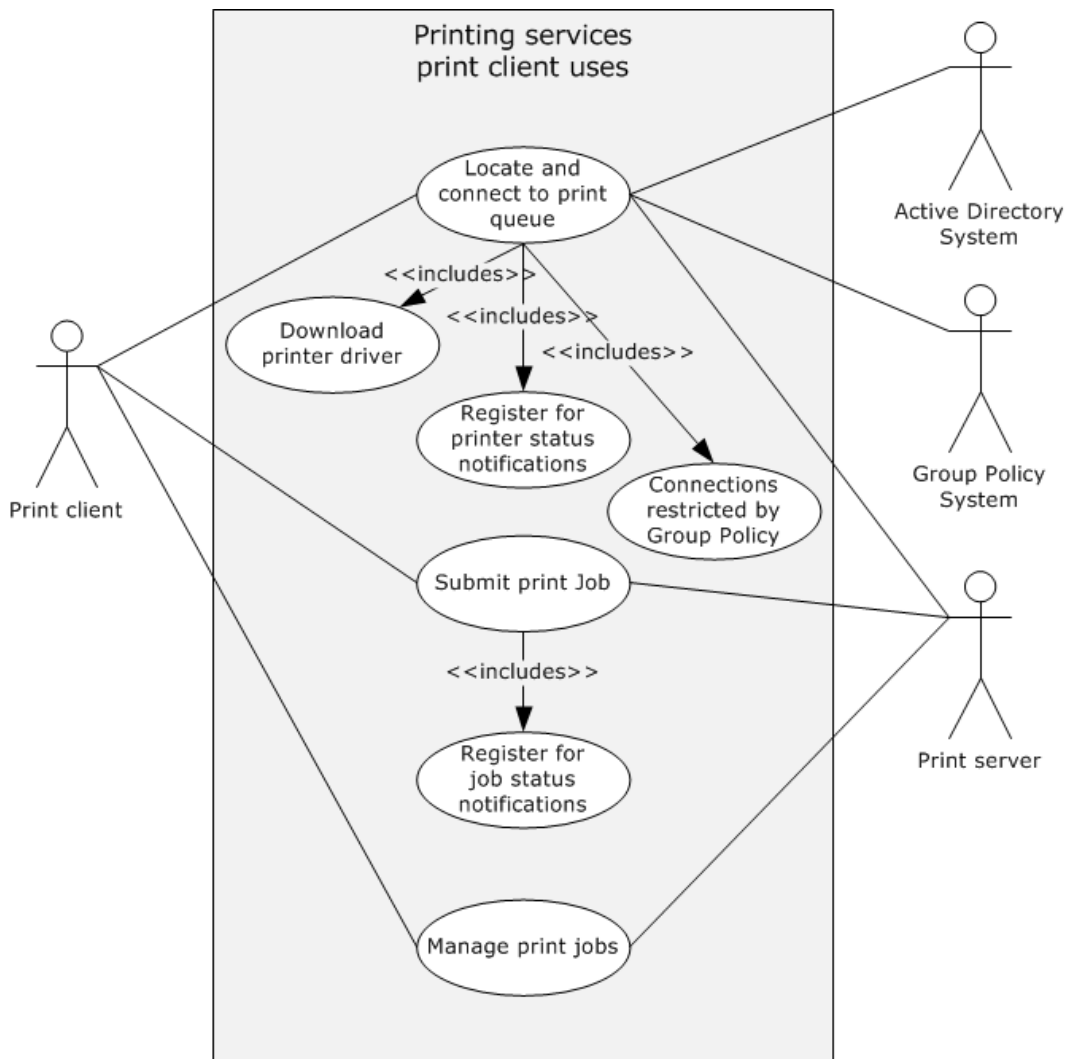
Figure 15: Printing services use cases initiated by a print client

2.5.3 Use Case Descriptions

2.5.3.1 Provision a Print Queue -- Administrative Client

Goal: To make a print queue available on a print server, subsequently allowing a user to select the print queue, establishing all attributes and components necessary to make the print queue discoverable and accessible to a print client.

Context of Use: Prior to a user being able to select a shared print queue to which to send a print job, an administrator is required to use an administrative tool to create and share a print queue on a print server. In Windows, this tool is the Print Management Console. This shared print queue corresponds to a printer. In addition to creating the print queue, attributes and components of the print queue are defined (such as the port to which the printer is connected, and the printer driver and print processor component that will convert the print job data to drawing commands that the printer uses). After an administrator provisions the print queue with the necessary information, a user using a print client (an application or operating system) can send a print job to the printer.

Direct Actor: The direct actor is the administrative client.

Primary Actor: The primary actor is an administrator.

Supporting Actors: The supporting actors are the print server and the Active Directory System.

Stakeholders and Interests:

▪ Administrative Client: The administrative client is used by an administrator to participate in this use case to add the necessary information to a print server for allowing a printer to be shared by multiple print clients.

▪ Administrator: An administrator participates in this use case by using an administrative client to configure a print server to share a printer among multiple users.

▪ Print Server: A print server is configured by this use case so that a printer connected to the print server can be shared among multiple users via a queue located on the print server.

▪ Active Directory System: The Active Directory System is updated by this use case so that a listing of the print queue is available via a print client when a user attempts to locate a print queue to which to send a print job.

Preconditions: Print server and administrative client are connected by using a network connection and both are members of the domain. The print spooler service is operational on the print server and administrative client. The Active Directory System is available and operational.

Minimal Guarantee: The administrative client is denied access and no provisioning operations occur.

Success Guarantee: A print queue corresponding to a printer is established on a print server and the Active Directory System is updated to allow multiple print clients to locate and share the printer.

Trigger: An administrator initiates the process of provisioning a print queue using an administrative tool to add a new print queue to a print server.

Main Success Scenario:

1. Using the SMB protocol family, the administrative client copies the files for the printer driver to a directory on the print server that is accessible via an SMB share.

2. The administrative client uses the protocol described in [MS-RPRN] to install the printer driver on the print server. The driver is associated with a print queue in a later step of this use case. In a network of print clients having different architectures, different versions of the same driver are uploaded and installed by the administrative client, allowing the print server to provide printer drivers to print clients whose architecture differs from the print server's architecture.

3. The administrative client ensures that an executable module for a port monitor is present on the print server (copying it there by using the SMB protocol family, if necessary), and then installs the port monitor on the print server using the protocol described in [MS-RPRN]. A port monitor implements and exports the methods that the print server calls locally to interface with a port to which a printer is connected. Ports for the port type implemented by the port monitor can only be added to the print server after the port monitor is added.

4. The administrator provides information about the port to which the printer is connected to the administrative client, which then uses the protocol described in [MS-RPRN] to direct the print server to add a specified printer port. A port is either a physical hardware port (such as a parallel

or USB port), or a network port (such as a WSD port or a TCP/IP port). The port is assigned to the print queue in a later step.

5. If the printer supports custom paper formats, called "forms", the administrative client can use the protocol described in [MS-RPRN] to add definitions for those custom forms to the print server. When added, such forms can be made available for use with the print queue.

6. The administrative client uses the protocol described in [MS-RPRN] to create a print queue on the print server. The administrator uses the administrative client to assign a printer driver, a printer port, and a print processor to the print queue during this step, and also sets access permissions, metadata (name of printer, physical location, and comments), time of availability, name under which the printer is shared, separator page settings, and processing priority.

7. The administrative client uses the protocol described in [MS-RPRN] to direct the print server to publish the print queue to the Active Directory System, which the print server satisfies by using the LDAP protocol [RFC4511] to create a directory object for the print queue, supplying the UNC name (server name plus share name) of the print queue. (Print clients can then use the LDAP protocol [RFC4511] to discover the print queue.)

Extension (a) - Install print processor on print server: Between steps 4 and 5 previously described, if the installed printer driver does not contain a print processor, the administrative client can use the protocol described in [MS-RPRN] to install a print processor on the print server.

Extension (b) – Enable Branch Office Print Mode: Between steps 6 and 7 previously described, the administrative client can use the protocol described in [MS-RPRN] to set properties on the print queue that control the Branch Office Print mode.

Variation (a) - Performing the use case using the protocol described in [MS-PAR]: All details identical to the use case described in this section except that the protocol described in [MS-PAR] is used to locate and establish the connection to the print queue.

2.5.3.2   Delete a Print Queue -- Administrative Client

Goal: To delete a previously provisioned print queue available on a print server, subsequently no longer allowing a user to select the print queue, removing all attributes and components necessary to make the print queue discoverable and accessible to a print client.

Context of Use: When a print queue is no longer needed, an administrator uses an administrative tool to delete a print queue from a print server. In Windows, this tool is the Print Management Console. After an administrator deletes the print queue, a user using a print client (an application or operating system) will no longer be able to send a print job to the printer.

Direct Actor: The direct actor is the administrative client.

Primary Actor: The primary actor is an administrator.

Supporting Actors: The supporting actors are the print server and the Active Directory System.

Stakeholders and Interests:

- Administrative Client: The administrative client will be used by an administrator to participate in this use case to delete a previously provisioned print queue.

- Administrator: An administrator will participate in this use case using an administrative client to configure a print server to delete a previously provisioned print queue.

- Print Server: A print server will be configured by this use case so that a printer connected to the print server can no longer be shared among multiple users via a queue located on the print server.

- Active Directory System: The Active Directory System will be updated by this use case so that the listing of the print queue will no longer be reported to a print client when a user attempts to locate a print queue to which to send a print job.

Preconditions: Print server and administrative client are connected by using a network connection and both are members of the domain. The print spooler service is operational on the print server and administrative client. The Active Directory System is available and operational. The print queue exists.

Minimal Guarantee: The administrative client is denied access and no delete operations occur.

Success Guarantee: A print queue is deleted from a print server and the Active Directory System is updated to no longer return the print queue in directory queries.

Trigger: An administrator initiates the process of deleting a print queue using an administrative tool to delete a print queue on a print server.

Main Success Scenario:

The administrative client uses the protocol described in [MS-RPRN] to direct the print server to unpublish the print queue from the Active Directory System, which the print server satisfies using the LDAP protocol [RFC4511] to delete the directory object for the print queue.

The administrative client uses the protocol described in [MS-RPRN] to delete the print queue on the print server.

Extension (a) - Deleting objects no longer referenced on print server: After deleting the print queue, the administrative client can use the protocol described in [MS-RPRN] to determine if the port, the print processor, and the printer driver previously used by the deleted print queue are still in use by other print queues; and if that is not the case, the administrative client can use the protocol described in [MS-RPRN] to delete said objects from the server.

Variation (a) - Performing the use case using the protocol described in [MS-PAR]: All details identical to the use case described in this section except that the protocol described in [MS-PAR] is used to delete the print queue.

2.5.3.3   Locate and Establish a Connection to a Print Queue in a Domain Environment -- Print Client

Goal: To make a connection to a shared print queue in a domain environment so documents can be printed by the user.

Context of Use: The user wants to print a document and does not have a printer attached to the local computer, but is member of a domain and knows that there are shared print queues in the enterprise available for use. The user decides to use one of the share print queues in the enterprise and initiates this use case.

Direct Actor: The direct actor is the print client.

Primary Actor: The primary actor is a user.

Supporting Actors: The supporting actors are print server, the Active Directory System, and Group Policy System (in Extension (b)).

Stakeholders and Interests:

▪ Print client: The print client will display a list of shared print queues from which a user will select a print queue, and then the print client will establish a connection to the selected print queue.

▪ Print server: The print server provides information about the shared print queue to the print client, makes the printer driver available to the print client for download if necessary, and allows the print client to connect to the print queue.

▪ Active Directory System: The Active Directory System store and provide a list of shared print queues that the user will view in the print client user interface.

▪ Group Policy System: The Group Policy Services Deployed Printer Connections Extension is a Supporting Actor in Extension (b). The Group Policy System is used to push a list of shared print queues to the print client so that the print client has pre-established connections to specified print queues. Additionally, Group Policy settings are inspected by the print client to restrict access to certain print servers.

▪ User: The user wants to view a list of available shared print queues and select one to which to send a print job.

Preconditions: The print spooler services are operational on the print client and print server. Both are members of the domain and connected with a network. The network is operational. The Active Directory System is available and operational.

Minimal Guarantee: The print queue is located but the user is informed via the print client user interface that a connection cannot be established due to insufficient permissions.

Success Guarantee: A connection to a shared print queue has been established by the print client and the connection can be used to submit print jobs to the print server.

Trigger: A user initiates this use case by starting the Add a printer command in the print client's user interface.

Main Success Scenario:

1. The print client queries the Active Directory System for a list of shared print queues in the domain.

2. The Active Directory System replies with a list of shared print queues.

3. The user of the print client selects a print queue from the query results.

4. The print client opens a handle to the selected print queue using the protocol described in [MS-RPRN].

5. The print client queries the print server for information about the print queue, such as which printer driver (and version) is being used, and the hardware ID of the printer represented by the print queue.

6. If the print client already has a copy of the printer driver installed, the print client creates a local print queue Proxy object representing the connection to the print queue on the print server (see Extension (a) for the case of when the print client does not already have a copy of the printer driver installed).

Extension (a) - Print client downloads printer driver:

Step 6: If the print client does not have a copy of the printer driver installed, the print client obtains the printer driver from one of the following sources, identifying the printer driver using the hardware ID of the print queue obtained earlier:

- From an administrative share on the print server

- From Windows Update Services

- From the user (for example, from a CD)

The print client then locally installs the printer driver.

Extension (b) - Print client makes connections only to print queues specified by Group Policy [MS-GPDPC] deployed printer connections:

Replace previous steps 1 through 3 with the following steps:

1. An administrator defines one or more deployed printer connection Group Policy settings for one or more domain users or user groups - each Group Policy setting includes details of one or more print queues shares by the print server.

2. Using the protocol described in [MS-GPDPC], the print client retrieves the deployed printer connections for the logged on user and the current computer and proceeds to automatically make connections to all print queues specified in the Group Policy settings, without user interaction (steps 4 through 6 for each print queue connection).

Extension (c) - Print client registers for notifications of print queue status:

Step 6 is followed by:

7. When the print queue Proxy representing the connection has been created, the print client opens a handle to the selected print queue using the protocol described in [MS-RPRN]. The print client then registers with the print server for some or all supported notifications on that handle, which will then be sent to the print client when available. The print client will reflect state changes of the server print queue in the local print queue Proxy object (such state changes can be, among others: error state, online/offline state, printer paused, list of queued jobs, associated printer driver - fetching and installing a different or updated printer driver if necessary, and so on.)

Extension (d) - Print client creates a Branch Office Printing version of the print queue:

Step 6 is followed by:

Step 7. When the print queue Proxy representing the connection has been created, the print client opens a handle to the selected print queue using the protocol described in [MS-RPRN]. The print client then requests the Branch Office Print configuration data from the print server for the print queue. If Branch Office Print Mode is enabled for this print queue, the print client will configure the print queue proxy to communicate directly with the device for print operations instead of with the print server.

Variation (a) - Performing the use case using the protocol described in [MS-PAR]: All details identical to the use case described in this section, except that the protocol described in [MS-PAR] is used instead of the protocol described in [MS-RPRN].

2.5.3.4   Locate and Establish a Connection to a Print Queue in a Workgroup Environment -- Print Client

Goal: To make a connection to a shared print queue in a workgroup environment so documents can be printed by the user.

Context of Use: The user wants to print a document and does not have a printer attached to the local computer, but is member of a workgroup and knows that there are shared print queues in the workgroup available for use. The user decides to use one of the shared print queues in the workgroup and initiates this use case.

Direct Actor: The direct actor is the print client.

Primary Actor: The primary actor is a user.

Supporting Actors: The supporting actors are print clients performing a print server role.

Stakeholders and Interests:

- Print client: The print client will display a list of shared print queues from which a user will select a Print queue, and then the print client will establish a connection to the selected print queue.

- Print clients performing a print server role: In this use case, the print server role is performed by Print clients that have shared print queues. After a user shares a print queue on the computer, the computer performs print server functions, including providing printer drivers to print clients, and accepting and printing print jobs from print clients. These print servers announce themselves to all other computers acting as print servers in the workgroup, as well as storing a list of all the other computers performing the print server role.

- User: The user wants to view a list of available shared print queues and select one to which to send a print job.

Preconditions: The print server has used the protocol described in [MS-BRWS] to find other computers (print servers and print clients) in the workgroup, and then used the protocol described in [MS-RPRN] to push the list of shared print queues to all print servers in the workgroup. The print server has also notified all other computers in the workgroup that it is a print server itself, using the protocol described in [MS-BRWS]. The print spooler service is operational on print server and print client. Both are connected with a network connection.

Minimal Guarantee: The print queue is located but the user is informed via the print client user interface that a connection cannot be established due to insufficient permissions.

Success Guarantee: A connection to a shared print queue has been established by the print client and the connection can be used to submit print jobs to the print server.

Trigger: A user initiates this use case by starting the Add a printer command in the print client user interface.

Main Success Scenario:

1. The user initiates the Add a printer action and selects the option to look for a printer.

2. The print client queries print servers in the workgroup for lists of shared print queues.

3. The user of the print client selects a print queue from presented list of shared print queues in the workgroup.

4. The print client opens a handle to the selected print queue using the protocol described in [MS-RPRN].

5. The print client queries the print server for information about the print queue, such as which printer driver (and version) is being used, and the hardware ID of the printer represented by the print queue.

6. If the print client already has a copy of the printer driver installed, the print client creates a local print queue proxy object representing the connection to the print queue on the print server (see Extensions for the case of when the print client does not already have a copy of the printer driver installed).

Extension (a) - Print client downloads a printer driver:

Step 4: After opening a handle to the selected print queue, if the print client does not have a copy of the printer driver installed, the print client downloads the printer driver from either (a) Windows Update, using the hardware ID of the print queue obtained earlier, or (b) from an administrative printer share of the print server (or other print client) in the workgroup. The print client then locally installs the printer driver before proceeding with the rest of the use case.

Extension (b) - Print client registers for notifications of print queue status:

Step 4. When the print queue representing the connection has been created, the print client opens a handle to the selected print queue using the protocol described in [MS-RPRN]. The print client then registers with the print server (or other print client) for some or all supported notifications on that handle, which will then be sent to the print client when available. The print client will reflect state changes of the server print queue in the local print queue object (such state changes can be, among others: error state, online/offline state, printer paused, list of queued jobs, associated printer driver - fetching and installing a different or updated printer driver if necessary, and so on.) When the print client is no longer interested in notifications (for example, because the user closes the user interface that shows status changes), the print client unregisters for notifications.

Variation (a) - Performing the use case using the protocol described in [MS-PAR]: All details identical to the use case described in this section except that the protocol described in [MS-PAR] is used instead of the protocol described in [MS-RPRN].

2.5.3.5   Locating and Connecting to a Shared Print Queue from an Internet Client -- Print Client

Goal: To make a connection to a shared print queue from an Internet client so documents can be printed by the user.

Context of Use: The user wants to print a document and does not have a printer attached to the local computer. The environment blocks connections using protocols described in [MS-RPRN] and [MS-PAR] (for example, via firewall). As an alternative, the Internet Printing Protocol ([RFC2911] and [RFC2910]) and the protocol described in [MS-WPRN] are available because they use HTTP/HTTPS and are typically allowed by most firewall administrators. Using these protocols, the print client allows the user to choose a shared printer via the Internet.

Direct Actor: The direct actor is the print client.

Primary Actor: The primary actor is the user.

Supporting Actors: The supporting actors are the print server and Internet Browser.

Stakeholders and Interests:

- Print client: The print client can make a connection to a print queue when the user enters the name of the print queue, as well as downloading a printer driver using the protocol described in [MS-WPRN]. The print client sends a print job to the print queue using the Internet Printing Protocol ([RFC2911] and [RFC2910]).

- Print server: The print server receives an HTTP GET request from an Internet Browser, generates a list of shared printers in HTML format, and responds to the HTTP GET request with the list to the print client. The print server also provides a printer driver to the print client.

- Internet Browser: The Internet Browser is used to connect to a print server and view shared print queues on the print server. The Internet Browser can also be used to connect the print client to a selected print queue by executing script code which calls local APIs of the print spooler. Alternately, the user can use the Internet Browser only for finding out the name of the print queue, and can then enter that name in the print client user interface to connect to the print queue.

- User: The user wants to send a print job to a print queue but is unable to use the print client to discover the shared print queues due to network restrictions.

Preconditions: The optional Internet Printing server role component has been installed on the print server. The print spooler service is operational on the print server and print client. An HTTP/HTTPS connection can be established between the print client and the print server. The user knows the URL of the print server.

Minimal Guarantee: A connection cannot be made.

Success Guarantee: A connection can be made.

Trigger: A user initiates this use case by starting the Add a printer command in the print client user interface.

Main Success Scenario:

1. The user of an Internet Browser browses to the "http://<HOST>/printers" URL of the print server that has the address <HOST>.

2. The print server returns an HTML page showing the available print queues and their status.

3. The user selects a print queue to which to connect.

4. The print client uses the protocol described in [MS-WPRN] to download the printer driver for the print queue from the print server.

5. The print client installs the downloaded printer driver.

6. The print client creates a local print queue Proxy object that represents the Internet Printing Protocol ([RFC2911] and [RFC2910]) connection to the print queue on the print server.

Extensions: None.

2.5.3.6 Setting Permissions for a Print Queue -- Administrative Client

Goal: To set permissions for a print queue, such as the priority of the print queue and the times it is available for shared use, as well as who can access it.

Context of Use: The administrator wants to restrict use of a shared print queue to a selected group of users or individual users. Additionally, the administrator wants to restrict the times of day when the shared print queue is available for use.

Direct Actor: The direct actor is the administrative client.

Primary Actor: The primary actor is the administrator.

Supporting Actors: The supporting actors are the print server and the Active Directory System.

Stakeholders and Interests:

- Administrative Client: The administrative client connects to the print server to send information regarding restrictions on the use of shared print queues.

- Print server: The print server restricts the use of shared print queues as specified by the administrative client.

- Administrator: The administrator wants to restrict the use of a shared print queue to a selected group of users.

Preconditions: The print server and administrative client are connected using a network connection and both are members of the domain. The print spooler service is operational on the print server and administrative client.

Minimal Guarantee: The administrative client denies the administrator the privilege to set permissions based on the administrator's authorization.

Success Guarantee: The permissions and time restrictions are set as requested.

Trigger: An administrator initiates the process of setting the permissions for a print queue by selecting a print queue using an administrative client.

Main Success Scenario:

1. The administrative client uses the protocol described in [MS-RPRN] to enumerate the print queues on the print server.

2. The administrator selects one of the enumerated print queues.

3. The administrative client uses the protocol described in [MS-RPRN] to open a handle to an existing print queue on the print server.

4. The administrative client uses the LDAP protocol [RFC4511] to obtain a list of domain users and groups from the Active Directory System.

5. The administrator selects users and groups from the list obtained in step 4 and assigns access permissions to form an access control list (ACL) and **security descriptor**.

6. The administrative client uses the protocol described in [MS-RPRN] to set the desired security descriptor on the print queue.

7. The administrator further chooses a time interval in which the print queue will accept print jobs from print clients.

8. The administrative client uses the protocol described in [MS-RPRN] to set the desired availability time interval on the print queue.

9. The administrative client closes the printer handle.

Extensions: None.

Variation (a) - Performing the use case using the protocol described in [MS-PAR]: All details identical to the use case described in this section except that the protocol described in [MS-PAR] is used instead of the protocol described in [MS-RPRN].

2.5.3.7   Submitting a Print Job -- Print Client

2.5.3.7.1   Submitting a Print Job Using the Protocols Defined in [MS-RPRN] (or [MS-PAR])

Goal: To print a document.

Context of Use: The user using an application capable of printing a document wants to print. A connection to a shared print queue has been previously established. The user initiates the Print command from the application.

Direct Actor: The direct actor is the print client.

Primary Actor: The primary actor is the user.

Supporting Actors: The supporting actor is the print server.

Stakeholders and Interests:

- Print client: The print client sends the print job data to the print queue on the print server.

- Print server: The print server buffers the print job data sent to the print queue to which the print client is connected, optionally processes the print job data further, and sends it to the printer associated with the print queue.

- User: The user desires a printed copy of content on the computer and chooses the Print function from the print client user interface.

Preconditions: The print spooler services are operational on the print client and print server. Both are members of the domain and connected with a network. The network is operational. The Active Directory System is available and operational.

Minimal Guarantee: The print job was submitted to the print server, but no job status feedback could be initiated, so the print job might print, but the print client cannot show feedback.

Success Guarantee: The job is submitted and job progress status was received and displayed to the user.

Trigger: A user initiates this use case by selecting the Print command from a printing capable application.

Main Success Scenario:

1. The print client opens a printer handle using the protocol described in [MS-RPRN].

2. The print client starts a new print job to the print server using the protocol described in [MS-RPRN].

3. The print client indicates the start of a new logical page to the print server, repeatedly sends data for the page, and signals the end of a logical page to the print server using the protocol described in [MS-RPRN]. The print client repeats this step for all pages in the document.

4. After sending all the pages of the print job to the print server, the print client ends the print job using the protocol described in [MS-RPRN].

5. The print client closes the printer handle using the protocol described in [MS-RPRN].

Extension (a) - Print client obtains notifications about print job status:

Following the previous step 1 and before step 2: The print client registers for change notifications using the protocol described in [MS-RPRN].

In parallel with the remaining steps, the print server sends change notifications as the processing of the print job proceeds. The print client provides feedback to the user.

Extension (b) - Print client obtains notifications for IHV-defined components on the print server and shows the user interface to the user:

Prior to the previous step 1, the print client uses the protocol described in [MS-RPRN] to register for change notifications from the print server.

In parallel with the other steps of the use case, when the print client receives change notifications signaling the arrival of a new print job, the print client will attempt to listen to the print queue for connection requests from the print server using the protocol described in [MS-PAN]. An IHV-defined printer driver or other IHV-defined component running on the print server can act as a notification source using the protocol described in [MS-PAN] and send unidirectional or bidirectional notification messages to the print client. One important notification message type for IHV-defined components is used to request that the print client shows a user interface to the user.

After step 4, if a connection has been established between the print client and the IHV-defined components using the protocol described in [MS-PAN], then the print client continues to listen for print server change notifications signaling the end of the print job for the monitored print queue. The print client will close the [MS-PAN] connection when all the jobs have been processed.

Variation (a) - Performing the use case using the protocol described in [MS-PAR]: All details identical to the use case described in this section except that the protocol described in [MS-PAR] is used instead of the protocol described in [MS-RPRN].

2.5.3.7.2 Submitting a Print Job by Using the Internet Printing Protocol

Goal: To print a document.

Context of Use: The user using an application capable of printing a document wants to print. A connection to a shared print queue has been previously established. The user initiates the Print command from the application.

Direct Actor: The direct actor is the print client.

Primary Actor: The primary actor is the user.

Supporting Actors: The supporting actors are the print server and a browser.

Stakeholders and Interests:

- Print client: The print client sends the print job data to the print queue on the print server.

- Print server: The print server buffers the print job data sent to the print queue to which the print client is connected, optionally processes the print job data further, and sends it to the printer associated with the print queue.

- User: The user desires a printed copy of content on the computer and chooses the Print function from the print client user interface.

Preconditions: The print spooler services are operational on the print client and print server. Both are members of the domain and connected with a network. The network is operational. The Active Directory System is available and operational.

Minimal Guarantee: The print job was submitted to the print server, but no job status feedback could be initiated, so the print job might print, but the print client cannot show feedback.

Success Guarantee: The job is submitted and job progress status was received and displayed to the user.

Trigger: A user initiates this use case by selecting the Print command from a printing capable application.

Main Success Scenario:

1. User initiates a print job from a Windows application to a print queue that is a connected to an Internet Printing Protocol ([RFC2911] and [RFC2910]) print queue on the print server.

2. The print client uses the Internet Printing Protocol ([RFC2911] and [RFC2910]) to submit the print job to the Internet Printing Protocol print queue on the print server.

3. The print client indicates the start of a new logical page to the print server, repeatedly sends data for the page, and signals the end of a logical page to the print server using the protocol described in [MS-RPRN]. The print client repeats this step for all pages in the document.

4. After sending all the pages of the print job to the print server, the print client ends the print job using the protocol described in [MS-RPRN].

5. The print client closes the printer handle by using the protocol described in [MS-RPRN].

Extensions: None.

2.5.3.7.3   Submitting a Print Job Using the SMB Protocol Family

Goal: To print a document.

Context of Use: The user wants to print a document by using an application capable of printing. A connection to a shared print queue has been previously established. The user initiates the Print command from the application.

Direct Actor: The direct actor is the print client.

Primary Actor: The primary actor is the user.

Supporting Actors: The supporting actor is the print server.

Stakeholders and Interests:

- Print client: The print client sends the print job data to the print queue on the print server.

- Print server: The print server buffers the print job data sent to the print queue to which the print client is connected, optionally processes the print job data further, and sends it to the printer associated with the print queue.

- User: The user wants a printed copy of content on the computer and chooses the Print function from the print client user interface.

Preconditions: The print spooler services are operational on the print client and print server. Both are members of the domain and connected with a network. The network is operational. The Active Directory System is available and operational.

Minimal Guarantee: The print job was submitted to the print server, but no job status feedback could be initiated, so the print job might print, but the print client cannot show feedback.

Success Guarantee: The job is submitted and job progress status was received and displayed to the user.

Trigger: A user initiates this use case by selecting the Print command from a printing capable application.

Main Success Scenario:

1. The user initiates a print job from the command line by executing a copy /b FILE \\SERVER\PRINTQ command (where FILE is a local file containing print job data, SERVER is the name of the server, and PRINTQ is the name of a shared print queue).

2. The print client uses the SMB protocol family to submit the file containing the print job data to a printer share on the print server.

3. The print client indicates the start of a new logical page to the print server, repeatedly sends data for the page, and signals the end of a logical page to the print server using the protocol described in [MS-RPRN]. The print client repeats this step for all pages in the document.

4. After sending all the pages of the print job to the print server, the print client ends the print job using the protocol described in [MS-RPRN].

5. The print client closes the printer handle using the protocol described in [MS-RPRN].

2.5.3.7.4 Submitting a Print Job Using Branch Office Print Mode

Goal: To print a document.

Context of Use: The user using an application capable of printing a document wants to print. A connection to a shared print queue has been previously established. The shared print queue is in branch office print mode. The user initiates the Print command from the application.

Direct Actor: The direct actor is the print client.

Primary Actor: The primary actor is the user.

Supporting Actors: The supporting actor is the print server.

Stakeholders and Interests:

- Print client: The print client sends the print job data directly to the print device.

- Print server: The print server might receive **branch office print remote logging** entries corresponding to the print job status from the print client.

- User: The user desires a printed copy of content on the computer and chooses the Print function from the print client user interface.

Preconditions: The print spooler service is operational on the print client. The print client is a member of the domain and connected with a network. The network is operational. The Active Directory System is available and operational.

Minimal Guarantee: The print job was submitted directly to the print device.

Success Guarantee: The job is submitted and job progress status was received and displayed to the user.

Trigger: A user initiates this use case by selecting the Print command from a printing capable application.

Main Success Scenario:

1. The print client renders the metafile to the printer-specific raw format before sending it as a data-stream payload directly to the print device.

2. The print client starts a new print job to the print device using either the TCPMON port monitor module (see [MS-RPRN] section 3.1.4.11.3) or WSDMON (Web Services for Devices Port Monitor, see [MS-RPRN] section 3.1.4.11.4).

3. After sending all the pages of the print job to the print device, the print client ends the print job.

Extension (a) - Print client sends branch office print remote logging entries to the print server:

Following the previous step 3:

4. The print client checks branch office print configuration data to see if branch office print remote logging is enabled.

5. If enabled, the print client opens a printer handle using the protocol described in [MS-RPRN].

6. The print client creates and sends applicable branch office print remote logging entries to the print server for processing using the protocol described in [MS-RPRN].

7. The print client checks the branch office print offline archive, and if populated sends the contained branch office print remote logging entries to the print server for processing using the protocol described in [MS-RPRN].

8. The print client closes the printer handle using the protocol described in [MS-RPRN].

Extension (b) - Print client archives branch office print remote logging entries for an offline print server:

Following the previous step 3:

4. The print client checks branch office print configuration data to see if branch office print remote logging is enabled.

5. If enabled the print client attempts to open a printer handle using the protocol described in [MS-RPRN].

6. If the print server is not available the print client creates the branch office print remote logging entries in the branch office print offline archive using the protocol described in [MS-RPRN].

Variation (a) - Performing the use case using the protocol described in [MS-PAR]: All details identical to the use case described in this section except that the protocol described in [MS-PAR] is used instead of the protocol described in [MS-RPRN].

2.5.3.8 Managing Print Jobs — Print Client

2.5.3.8.1 Managing Print Jobs Submitted by Self Using the Protocols Described in [MS-RPRN] or [MS-PAR]

Goal: For a user to manage his or her own submitted print jobs, including pausing them, resuming them, canceling them, changing their priority, changing their order in the queue, or restarting them.

Context of Use: After submitting a print job, a user might want to manage the job for a variety of reasons. For example, a user might want to pause a job in order to print another job of higher priority, or a user might cancel a job when it becomes apparent that the content is flawed and printing would only waste paper.

Direct Actor: The direct actor is the print client.

Primary Actor: The primary actor is the user.

Supporting Actors: The supporting actor is the print server.

Stakeholders and Interests:

- Print client: The Print client is used to display the print jobs submitted by the user, allow the user to select a print job to manage, and allow the user to select management functions.

- Print server: The print server receives and executes job management functions requested from a print client or administrative client.

- User: A user wants to manage the printing of a print job, pausing, canceling, or changing the priority of the print job, for example.

Preconditions: The print spooler services are operational on the print client and print server. Both are members of the domain and connected with a network. The network is operational.

Minimal Guarantee: The user or administrator attempting to perform management functions on a print job will be denied permission via the user interface on the print client or the administrative client.

Success Guarantee: The user or administrator attempting to perform management functions on a print job will successfully be able to pause a job, resume a job, cancel a job, or perform other functions.

Trigger: The user opens the print queue user interface and looks at queued jobs prior to modifying them (cancel, reorder, pause, resume, and so on).

Main Success Scenario:

1. The print client opens a printer handle by using the protocol described in [MS-RPRN].

2. The print client enumerates jobs scheduled for printing on the printer using the protocol described in [MS-RPRN].

3. The print client opens a handle to a specific job using the protocol described in [MS-RPRN]. The print server will refuse this request for any job not submitted by the user of the print client.

4. The print client modifies job settings or job priority using the protocol described in [MS-RPRN].

5. The print client closes the printer handle using the protocol described in [MS-RPRN].

Variation (a) - Performing the use case by using the protocol described in [MS-PAR]: All details identical to the use case described in this section except that the protocol described in [MS-PAR] is used instead of the protocol described in [MS-RPRN].

2.5.3.8.2 Managing Print Jobs Submitted by All Users Using the Protocols Described in [MS-RPRN] or [MS-PAR]

Goal: For an administrator to manage print jobs submitted by all users, including pausing, resuming, canceling, changing their priority, changing their order in the queue, or restarting.

Context of Use: An administrator might need to override or reconfigure print jobs that various users have submitted to multiple print queues in order to effectively manage or maintain printing resources. An administrator can use an administrative client or a print client to manage print jobs submitted by all users.

Direct Actor: The direct actor is the administrative client or print client.

Primary Actor: The primary actor is the administrator.

Supporting Actors: The supporting actor is the print server.

Preconditions: An administrator possesses administrative permissions for the print queue, enabling the print server to permit the request to open a job handle of any job, regardless of who submitted the job.

Stakeholders and Interests:

- Print client: The print client is used to display the print jobs submitted by the user, allow the user to select a print job to manage, and allow the user to select management functions.

- Administrative client: The administrative client is used to display the print jobs submitted by all users, allow the administrator to select print jobs to manage, and allow the administrator to select management functions.

- Print server: The print server receives and executes job management functions requested from a print client or administrative client.

- User: A user wants to manage the printing of a print job, pausing, canceling, or changing the priority of the print job, for example.

- Administrator: An administrator wants to manage all the print jobs that have been submitted to a print queue in order to service the printer, free up the printer for alternate uses, or to intervene when discovering that a very large print job has been sent to a printer not capable of handling such volume.

Preconditions: An administrator possesses administrative permissions for the print queue, enabling the print server to permit the request to open a job handle of any job, regardless of who submitted the job. The print spooler services are operational on the print client and print server. Both are members of the domain and connected with a network. The network is operational.

Minimal Guarantee: The user or administrator attempting to perform management functions on a print job will be denied permission via the user interface on the print client or the administrative client.

Success Guarantee: The user or administrator attempting to perform management functions on a print job will successfully be able to pause a job, resume a job, cancel a job, or perform other functions.

Trigger: The user opens the print queue user interface and looks at queued jobs prior to modifying them (cancel, reorder, pause, resume, and so on).

Main Success Scenario:

1. The print client opens a printer handle by using the protocol described in [MS-RPRN].

2. The print client enumerates jobs scheduled for printing on the printer by using the protocol described in [MS-RPRN].

3. The print client opens a handle to a specific job using the protocol described in [MS-RPRN]. The print server will refuse this request for any job not submitted by the user of the print client.

4. The print client modifies job settings or job priority by using the protocol described in [MS-RPRN].

5. The print client closes the printer handle by using the protocol described in [MS-RPRN].

Variation (a) - Performing the use case using the protocol described in [MS-PAR]: All details identical to the use case described in this section except that the protocol described in [MS-PAR] is used instead of the protocol described in [MS-RPRN].

2.5.3.8.3   Managing Print Jobs from a Command Line Using the Protocol Described in [MS-RAP]

Goal: For a user to manage his or her own submitted print jobs, including pausing, resuming, canceling, changing their priority, changing their order in the queue, or restarting.

Context of Use: After submitting a print job, a user might need to manage the job for a variety of reasons. For example, a user might want to pause a job in order to print another job of higher priority; or a user might cancel a job when it becomes apparent that the content is flawed and printing would only waste paper.

Direct Actor: The direct actor is the print client.

Primary Actor: The primary actor is the user.

Supporting Actors: The supporting actor is the print server.

Stakeholders and Interests:

- Print client: The print client is used to display the print jobs submitted by the user, allow the user to select a print job to manage, and allow the user to select management functions.

- Print server: The print server receives and executes job management functions requested from a print client or administrative client.

- User: A user wants to manage the printing of a print job, pausing, canceling, or changing the priority of the print job, for example.

Preconditions: The print spooler services are operational on the print client and print server. Both are members of the domain and connected with a network. The network is operational.

Minimal Guarantee: The user or administrator attempting to perform management functions on a print job will be denied permission via the user interface on the print client or the administrative client.

Success Guarantee: The user or administrator attempting to perform management functions on a print job will successfully be able to pause a job, resume a job, cancel a job, or perform other functions.

Trigger: The user opens the print queue user interface and looks at queued jobs prior to modifying them (cancel, reorder, pause, resume, and so on).

Main Success Scenario:

1. The user uses the net print command to enumerate print jobs on the print queue.<4>

2. The print client uses the protocol described in [MS-RAP] to enumerate print jobs on the print queue of the print server.

3. The user uses the net print command to pause, resume or cancel print jobs on the print queue.<5>

4. The print client uses the protocol described in [MS-RAP] to pause, resume, or cancel a print job.

2.6 Versioning, Capability Negotiation, and Extensibility

The Print Services System (PSS) evolved over multiple releases of Windows client and server products, namely Windows NT 3.5 operating system, Windows NT Server 3.51 operating system, Windows NT 4.0 operating system, Windows 2000 operating system, Windows XP operating system, Windows Server 2003 operating system, Windows Vista operating system, Windows Server 2008 operating system, Windows 7 operating system, Windows Server 2008 R2 operating system, Windows 8 operating system, Windows Server 2012 operating system, Windows 8.1 operating system, and Windows Server 2012 R2 operating system. Each version of the Print Services System provides additional functionality, while maintaining interoperability with previous versions. For the purposes of this document, the Print Services System functionality is grouped using the following versions:

- PSS 1.0 (Windows 3x, Windows 9x, Windows Millennium Edition operating system, Windows NT operating system)

- PSS 2.0 (Windows NT)

- PSS 3.0 (Windows 2000, Windows Server 2003, Windows XP, Windows Server 2008)

- PSS 4.0 (Windows Vista, Windows Server 2008 R2, Windows 7, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2)

The functionality in these versions is incremental. In order for a third party print server to interoperate with a Windows print client, the print server has to support all the functionality for a given version and the versions that preceded it.

For example, if PSS 3.0 included a protocol defining opnums 1 through 10, and PSS 4.0 included a new protocol as well as an extension to the PSS 3.0 protocol defining a new opnum 11, then Windows print clients require that the print server supports the new PSS 4.0 protocol, as well as the opnum 11 extension to the PSS 3.0 protocol.

Support for some protocols is optional, as shown in [Member Protocol Functional Relationships](#), section [2.1.2.3](#), and as described later in this section. An implementation can be fully functional in a typical Windows network without implementing the optional protocols.

The print client or administrative client determines the print server's PSS version support by using the following sequence of operations, calling the methods described in [MS-RPRN]:

1. The print client calls the **RpcOpenPrinter** method on the print server by using the print server's name to obtain a handle.

2. The print client calls the **RpcGetPrinterData** method on the print server by using the OSVersion key to obtain the print server's operating system version (and maps the OSVersion of the print server to the supported PSS version).

3. The print client calls the **RpcClosePrinter** method on the print server to close the print server handle.

In addition to following the process previously described to determine the capabilities of the print server, print clients also follow the processes defined by the protocols of the Print Services System for negotiating capability support.

The following tables describe how the protocols used in the Print Services System are supported by the different versions of the system. The Windows product releases that correspond to each version of the Print Services System are listed in the [Product Behavior Appendix](#).[<6>](#)

Microsoft-defined protocols, standard protocols, and print payloads are described in the following tables.

PSS 1.0

Protocols supported by print server	Print payloads supported by print server	Protocols used by print client	Print payloads used by print client	Protocol notes
RAP, SMB, Line Printer Daemon Protocol, and CIFS	Raw	SMB and Line Printer Daemon Protocol	Raw	Line Printer Daemon Protocol (support optional): Print servers provide support for this protocol for interoperability with UNIX clients. Print clients use this protocol only if specifically configured to do so when SMB is blocked by a firewall.

PSS 2.0

Protocols supported by print server	Print payloads supported by print server	Protocols used by print client	Print payloads used by print client	Protocol notes
PSS 1.0 (RAP, SMB, Line Printer Daemon Protocol, and CIFS)	PSS 1.0 (Raw) plus EMFSPOOL	PSS 1.0 (SMB and Line Printer Daemon Protocol)	PSS 1.0 (Raw) plus EMFSPOOL	Line Printer Daemon Protocol (support optional): Print servers provide support for this protocol for interoperability with UNIX clients. Print clients support this protocol only if specifically configured to do so when SMB protocol and Print System Remote Protocol are

Protocols supported by print server	Print payloads supported by print server	Protocols used by print client	Print payloads used by print client	Protocol notes
plus Print System Remote Protocol		plus Print System Remote Protocol		blocked by firewall.
				SMB: Print clients use this protocol for driver file download operations or if Print System Remote Protocol is not available on a print server.
				EMFSPOOL (support optional): Print clients can use this payload format only if the print server advertises support through the **RPRN::RpcEnumPrintProcessorDatatypes** method.
				Print System Remote Protocol: This protocol is supported in different versions of Windows as indicated in Product Behavior Appendix.

PSS 3.0

Protocols supported by print server	Print Payloads supported by print server	Protocols used by print client	Print payloads used by print client	Protocol notes
PSS 2.0 (RAP, SMB, Line Printer Daemon Protocol, and Print System Remote Protocol) plus Web Point-and-Print Protocol, Internet Printing Protocol, Group Policy: Deployed Printer Connections Protocol Extension, LDAP (v3) and Group Policy: Core Protocol	PSS 2.0 (Raw and EMFSPOOL)	PSS 2.0 (SMB, Line Printer Daemon Protocol, and Print System Remote Protocol) plus Web Point-and-Print Protocol, Internet Printing Protocol, Group Policy: Deployed Printer Connections Protocol Extension, LDAP, and Group Policy: Core Protocol	PSS 2.0 (Raw, EMFSPOOL)	Line Printer Daemon Protocol (support optional): Print servers provide support for this protocol for interoperability with UNIX clients. Print clients support this protocol only if specifically configured to do so when SMB protocol and Print System Remote Protocol are blocked by firewall.

Web Point-and-Print Protocol (support optional): print servers support is protocol to provide printer drivers for print clients connecting through Internet Printing Protocol. Print clients use this protocol to download printer drivers from a print server when using the Internet Printing protocol to connect.

Internet Printing Protocol (support optional): Print servers provide support for this protocol to allow printing from print clients on the extranet via HTTP. Print clients use this protocol only if specifically requested to connect using an HTTP printer connection, which is done when a firewall blocks the other printing protocols.

Group Policy: Deployed Printer Connections Protocol Extension (support optional): Support for this protocol by print servers and print clients was provided in PSS 3.0 by a downloadable utility.

SMB: Print clients use this protocol for driver file download operations or if Print System |

Protocols supported by print server	Print Payloads supported by print server	Protocols used by print client	Print payloads used by print client	Protocol notes
				Remote Protocol is not available on a server. EMFSPOOL (support optional): print clients can use this payload format only if the print server advertises support through the **RPRN::RpcEnumPrintProcessorDatatypes** method.

PSS 4.0

Protocols supported by print server	Print payloads Supported by print server	Protocols used by print client	Print payloads used by print client	Protocol notes
PSS 3.0 (RAP, SMB Version 1.0, SMB Version 2.0, Line Printer Daemon Protocol, Print System Remote Protocol, Web Point-and-Print Protocol, Internet Printing Protocol, Group Policy: Deployed Printer Connections Protocol Extension) and Print System Asynchronous Notification Protocol, Print System Asynchronous Remote Protocol, LDAP (v3), Group Policy: Core	PSS 3.0 (Raw, EMFSPOOL), XML Paper Specification	PSS 3.0 (SMB, Line Printer Daemon Protocol, Print System Remote Protocol, Web Point-and-Print Protocol, Internet Printing Protocol, Group Policy: Deployed Printer Connections Protocol Extension, LDAP, Group Policy: Core Protocol) and Print System Asynchronous Notification Protocol, Print System Asynchronous Remote Protocol	PSS 3.0 (Raw, EMFSPOOL), XML Paper Specification	Line Printer Daemon Protocol (support optional): Print servers provide support for this protocol for interoperability with UNIX clients. Print clients use this protocol only if specifically configured to do so when SMB protocol and Print System Remote Protocol are blocked by a firewall. Web Point-and-Print Protocol (support optional): Print servers support this protocol to provide printer drivers for clients connecting through the Internet Printing Protocol. Print clients use this protocol to download printer drivers from a print server when using the Internet Printing Protocol to connect. Internet Printing Protocol (support optional): Print servers provide support for this protocol to allow printing from print clients on the extranet via HTTP. Print clients use this protocol only if specifically requested to connect through an HTTP printer connection, which is done when a firewall blocks the other printing protocols. XML Paper Specification (support optional): Print servers support this payload format only with third-party components installed, or with Terminal Server Easy Print installed. SMB: Print clients use this protocol for driver file download operations, or if neither Print System Remote Protocol nor Print System Asynchronous Remote Protocol is available on a print server. Print System Remote Protocol: Windows Vista, Windows 7, Windows 8, and Windows 8.1, print clients use this protocol to check whether Print System Asynchronous Remote Protocol is available on the print server (by comparing the

Protocols supported by print server	Print payloads Supported by print server	Protocols used by print client	Print payloads used by print client	Protocol notes
Protocol				server's OS build number retrieved by using Print System Remote Protocol methods and to confirm a printer's presence by using a single Print System Remote Protocol **RpcOpenPrinter**/**RpcClosePrinter** method sequence), and then uses Print System Asynchronous Remote Protocol subsequently.
				Windows Vista, Windows 7, Windows 8, and Windows 8.1, print clients also use Print System Remote Protocol if Print System Asynchronous Remote Protocol is unavailable on the print server.
				EMFSPOOL (support optional): Print clients can use this payload format only if the print server advertises support through the **RPRN::RpcEnumPrintProcessorDatatypes** or **PAR::RpcAsyncEnumPrintProcessorDataty pes**.
				RAP was removed in Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2, because its functionality was being provided in other ways.

Vendor-specific extensions are supported as noted in section 1.8 of the member protocol specification Technical Documents (TDs).

2.7 Error Handling

Status and error returns are defined by member protocols of the Print Services System. The Print Services System does not define status codes or error codes in addition to those described in the technical documents relating to the member protocols.

2.7.1 Failure Scenarios

The following failure scenarios are discussed in this section:

- Abnormal termination of the print spooler service

- Loss of network connectivity

- Loss of system disk storage

- Print spooler service out-of-system resources

- Authentication issues

- Expected failures

2.7.1.1 Abnormal Termination of the Print Spooler Service

This problem is typically caused by a malfunctioning plug-in module, such as an IHV-supplied printer driver. Other causes include internal malfunction of the print spooler service and sudden loss of system power. The print spooler service terminates immediately without orderly shutdown. The system attempts to restart the print spooler service multiple times before giving up, at which point administrative intervention locally on the system experiencing the failure is required to diagnose the problem and take corrective action.

All print jobs completely submitted at the time of failure remain queued and their processing is restarted from the beginning upon print spooler restart.

Print clients are not informed of an abnormal termination event, and experience failures on currently executing and all subsequent calls made through member protocols. Print clients implement appropriate RPC rundown handlers to free RPC resources (handles) that are in use at the time of failure. Print clients retry failed calls until the print server is available again.

In a system configuration in which printer drivers are run by the print spooler in the print spooler's address space, an abnormal termination or memory corruption in a printer driver can abnormally terminate the print spooler process.

In a system configuration in which printer drivers are run in isolation processes (commonly called a sandbox),<7> a malfunctioning printer driver causes only the isolation process to terminate. Print clients are not informed of such a failure. All current completely submitted print jobs for all printer drivers sharing the terminated isolation process remain queued and their processing is restarted from the beginning upon restart of the isolation process. The print spooler attempts to restart the isolation process for a malfunctioning printer driver two times before giving up, in which case administrative intervention locally on the system experiencing the failure is required to diagnose the problem and take corrective action.

2.7.1.2 Loss of Network Connectivity

In case of connectivity loss, the protocol handler for each member protocol tries independently to reconnect. There is no system-wide detection of connectivity loss.

2.7.1.3 Loss of System Disk Storage

Loss of system disk storage is a fatal, non-recoverable event. The behavior of the system in this case is undefined and unpredictable. The print spooler service of the affected system is restarted manually when system storage comes back online.

2.7.1.4 Print Spooler Service Out of System Resources

Method calls to member protocols return a nonzero error code if the server cannot process the request due to lack of system resources (such as memory, handles, or disk space). There is no system-wide mechanism for dealing with a server that is out of system resources.

2.7.1.5 Authentication Issues

Authentication is handled independently for each component protocol. Credential storage on Windows makes a distinction between session-based transports (RPC over SMB) and non-session-based transports (RPC over TCP/IP). If secondary credentials have been added by the user or another subsystem to the credential manager, these will only be used for protocol requests using session based transport, and be unavailable for non-session based transport.

Specifically, that means that requests over protocols defined in [MS-PAR] and [MS-PAN] can fail due to failed authentication even if requests over the protocol defined in [MS-RPRN] operate correctly. Print client architecture is designed to handle these failures gracefully, and fall back to using the protocol defined in [MS-RPRN] if the protocols defined in [MS-PAR] or [MS-PAN] fail due to authentication issues. As a consequence, functionality specific to the protocols defined in [MS-PAR] or [MS-PAN] can be unavailable in such a fallback scenario.

2.7.1.6 Expected Failures

A group of failures caused by malfunctioning printers are considered expected failures and the member protocols contain facilities to report the type of failure to the print client. The print client responds to failures typically in an implementation-specific manner, such as by retrying (for example, in a printer offline scenario) and offering visual feedback to the user. The user can then make a choice of further action, such as canceling the job and reprinting to a different print queue, or walking up to the printer and fixing a paper jam.

Another kind of expected failure is that a buffer proves too small to receive data returned by an RPC call.

2.8 Coherency Requirements

2.8.1 Timers

If the print server is configured and Active Directory is available, the print server performs the following operations:

- Periodically enumerate through the **List of Printers** and update the Active Directory as described in [MS-RPRN] (section 2.3.3.4).

- Periodically search for the print queues in Active Directory (as described in [MS-RPRN] (section 2.3.3.3) and delete all print queues that are not present in the current **List of Printers** (as described in [MS-RPRN] (section 2.3.3.2)).

2.8.2 Non-Timer Events

When a system shutdown event occurs:

- The Print Services System closes all open handles to remote objects.

- The Print Services System unregisters endpoints.

When a local Plug and Play event occurs:

- When new printer hardware is detected on a local port of a computer acting as print server, the Print Services System automatically locates a matching driver for the detected printer and installs a print queue. The created print queue can optionally be set up as a shared print queue (per policy setting).

2.8.3 Initialization and Reinitialization Procedures

Initialization occurs at system startup, which starts the print spooler service.

The print spooler service listens on endpoints for protocols in the following order:

- At any time: Start listening to the Line Printer Daemon Protocol [RFC1179], which is done in a separate Windows service, LPDSVC. In Windows, this is done by Internet Information Services (IIS).

- At any time: Start listening to Web Point-and-Print Protocol [MS-WPRN] and the Internet Printing Protocol ([RFC2911] and [RFC2910]) simultaneously, or in the listed order. In Windows, this also is done by IIS.

- At any time: Start listening to Print System Asynchronous Notification Protocol [MS-PAN], Print System Asynchronous Remote Protocol [MS-PAR], and Print System Remote Protocol [MS-PAR] simultaneously, or in the listed order.

Reinitialization occurs up to two times only upon abnormal termination of the print spooler service.

The Print Services System starts up automatically when the server hosting the Print Services System boots. The Print Services System stays in the running state until a shutdown of the print server occurs, or it is manually stopped by an administrator.

The Print Services System persists elements of the ADM in the system registry.

On initialization, print services instantiate in-memory objects according to persisted information. In Windows 8 operating system, Windows Server 2012 operating system, Windows 8.1 operating system, and Windows Server 2012 R2 operating system, this does not occur until after conditions are met that show the machine to be fully ready to use (for example, after a user logs on, or after a significant time has elapsed since start-up).

2.9 Security

This section documents system-wide security issues that are not otherwise described in the Technical Documents (TDs) for the Member Protocols. It does not duplicate what is already in the Member Protocol TDs unless there is some unique aspect that applies to the system as a whole.

Windows Print Services allow installation of printer drivers on a print client from local storage media, or via copying the drivers from the print server for the shared printer. Windows printer drivers contain executable modules which are loaded and executed in the local system account by the print spooler.

As another mitigating tactic, a system administrator can restrict printer driver installation by the application of Group Policy settings accepted by the print client.

Print Services secures access to print queues and ports shared by a print server using access control lists (ACLs). However, printers on the network connected via the TCPMON monitor module (see [MS-RPRN] section 3.1.4.11.3) or WSDMON (Web Services for Devices Port Monitor, see [MS-RPRN] section 3.1.4.11.4) do not offer a mechanism for access protection; therefore, print clients can bypass the print server's protection mechanism by creating a direct TCPMON or WSDMON connection to a printer. The address of the printer can be determined by scanning the local network for ports used by TCPMON or WSDMON connections. Often, it is also possible to find printers by examining **Domain Name System (DNS)** entries for the local network, or by printing a printer status sheet on the printer itself (the status sheet typically shows the DNS name or the TCP/IP address of the printer).

Because of these limitations, in a Print Services installation with heightened access security requirements for printers it is recommended to connect the printers to a private network only accessible by the print server. In such a configuration, the print server always acts as an intermediary between the print client and the printer, and therefore can effectively enforce access security.

An additional security concern is the actual security and accessibility of the physically printed pages resulting from a print job. The Print Services System does not offer any solution for this problem; however, multiple IHV specific solutions exist, in which the IHV provided printer driver running on the print client will embed a PIN only known to the user into the IHV-specific command stream sent to the printer. The printer will only print or release the physical print job after the user is identified with the PIN at the printer.

Another security concern is that the member protocols of the Print Services System do not require the use of heightened transport security (as invoked using the RPC_C_AUTHN_LEVEL_PKT_INTEGRITY or RPC_C_AUTHN_LEVEL_PKT_PRIVACY authentication-level constants). Consequently print job data and other data transmitted using member protocols cannot be relied upon to be secure; such data can potentially be snooped on or tampered with. A secure print environment is achieved by employing the underlying transport security mechanisms, such as **Internet Protocol security (IPsec)**.

2.9.1 Security and Authentication Methods

Versioning of security is handled by the underlying RPC transport; see [MS-RPCE] section 3.3.3.3 for more information.

Policy settings controlling local print spooler behavior can be pushed by using the Group Policy: Core Protocol [MS-GPOL] from the directory server to print clients and print servers.<8> This functionality allows the following:

- Driver installation security settings

- Sandbox grouping settings

2.9.2 Securable Objects

The securable objects in the Print Services System are the print job object, print queue object, and print server object. Security for these objects is discussed in [MS-RPRN], [MS-PAR], and [MS-PAN].

Print services apply ACLs to elements of the ADM for the member protocols. In a typical managed network, default ACLs allow users the Use access right for all elements accessible through the member protocols, and allow administrators unrestricted access for all elements accessible through the member protocols.

2.9.3 Internal Security

The Print Services System applies ACLs to elements in its ADM. When processing calls, the system impersonates the calling user and verifies access to a secured element through the **authenticated user identity** (see [MS-PAN] section 3.1.1.6.5).

2.9.4 External Security

The Print Services System impersonates the user when processing calls. The permissions in the user's token determine the type of access this system has to external resources while processing calls.

Print clients and print servers for print services need to take particular care when printer drivers or other plug-ins are referenced on a remote system, or copied from a remote system. Printer drivers or other plug-ins can contain executable code, and therefore need to be constrained to execute only in the context of the local caller if they are trusted.

The Windows print spooler calls printer drivers or other plug-ins as local system calls and impersonates the calling user. As a consequence, the print spooler needs to take appropriate precautions to protect the system from harm by untrusted printer drivers.

The Windows implementation can perform one or more of the following actions:

- Restrict non-administrative users from installing printer drivers.

- Check the digital signatures of drivers.

- Prompt the user for consent before downloading such components or executing the component for the first time.

2.10   Additional Considerations

None.

3 Examples

The following examples illustrate the protocols used by the Print Services System to perform the following:

- Discover and utilize a print queue in a domain.

- Discover and utilize a print queue in a workgroup.

- Locate a print queue in a domain and establish a connection, then submit a print job to a manual duplex printer by using the protocol defined in [MS-PAN] and enable bidirectional IHV-defined communication between the print client and the print server.

- Enumerate print jobs from all users, then cancel several print jobs.

- Provision a print queue by using the protocol defined in [MS-RPRN] from an administrative client, then delete the same print queue by using the protocol defined in [MS-PAR] from a different administrative client.

- Send a print job to an SMB share.

In all examples discussed in this section, the print client opens, closes, and reopens the handle to the print queue multiple times. This is because the Windows API implementation for the print spooler allows and uses different processes for the print client user interface, print client applications, and internal features of the print spooler. Each of these processes can have multiple threads (for example, an application can have one thread printing in the background and another thread computing layout in the foreground). Other queries to the print server (for example, about print jobs) could happen in parallel from different threads, causing additional exchanges unrelated to a specific scenario on the wire.

In all examples discussed in this section, the print client retrieves information from the print server in arbitrary order, and often retrieves the same information redundantly. Methods that could be called in arbitrary order and redundantly are marked with an asterisk (*) in the examples.

Because handles are not generally portable between threads (and are never portable between processes), the print client could also use different handles to the same print queue for different functions, such as listening for notifications and submitting a print job.

3.1 Example 1: Discovering and Utilizing a Print Queue in a Domain

This example demonstrates the use cases described in section 2.5.3.3 with extensions (a) and (c), and section 2.5.3.7.1 with extension (a).

Prior to the beginning of this example, an administrator has installed a printer on a print server and provisioned a print queue corresponding to the printer. The print server has published the name of the print queue to the directory service by using the LDAP protocol [RFC4511]. In this scenario, PSS uses LDAP to discover the available shared print queues, after which it uses the Print System Remote Protocol [MS-RPRN] and the SMB Access Protocols for subsequent operations.

Prerequisites

The example described in this section and subsections has the following prerequisites:

- The print client communicates with the print server using the Print System Remote Protocol, which includes support for **RpcGetPrinterDriverPackagePath**.<9>

- The print queue has been provisioned and is located on a print server in a domain.

- The print server has published a list of the shared print queues to the Active Directory System.

- The print client is configured not to use Windows Update to obtain the printer driver. (This precondition is only present to illustrate downloading a printer driver from the print server.)

- The printer driver is available from the printer server and the print driver is enabled as package aware.<10>

- Either the print client or print server supports the Print System Remote Protocol [MS-RPRN], but not the Print System Asynchronous Remote Protocol [MS-PAR].<11> This precondition was chosen to illustrate the use of the Print System Remote Protocol.<12>

Initial System State

The example described in this section and subsections applies in the following initial system state:

- The user of the print client does not know the name of the print server and the print client does not yet have a list of the available print servers or print queues.

- The print client does not have the printer driver for the print queue.

- The print client does not have the printer **driver package** for the printer driver in its local **driver store**. (This state is present to illustrate a printer driver download from the print server.)

Final System State

- The print client obtains the list of available printers and print servers.

- The print client has downloaded a printer driver for the print queue.

This example is divided into four tasks:

1. Locating print queue in a domain

2. Establishing a connection, downloading a driver, and registering for notifications

3. Submitting a print job and receiving notifications

4. Unregistering for notifications

Task 1: Locating a print queue in a domain

The following diagram illustrates the first part of this example in which the print client locates a print queue in a domain by using LDAP.
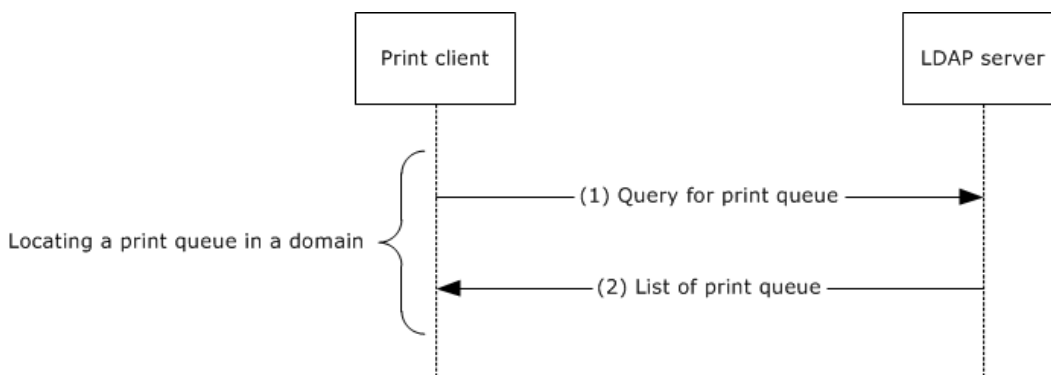
Figure 16: Print client locating a print queue in a domain by using LDAP

The message group labeled "Locating a print queue in a domain" in the preceding figure represents the search for a print queue in the Active Directory using the LDAP search option, as described in [RFC4511] section 4.5, and [MS-RPRN] (section 2.3.3.3).

The following steps describe task 1 of this example.

1. The print client uses LDAP to query the LDAP server for print queues, optionally specifying criteria, such as functionality, that the print queues are required to satisfy ([MS-RPRN] section 2.3.3.3). The print client obtains criteria from the user interface and makes several queries to obtain criteria, such as the physical location of the print client.

2. The LDAP server returns a list of print queues by using LDAP. The print client user interface then presents the list to the user, who selects a print queue.

   Task 2: Establishing a connection, downloading a driver, and registering for notifications

   The following diagram illustrates a print client connecting to a print queue, downloading a driver, and registering for notifications. The sequence is described in the steps that follow the diagram.
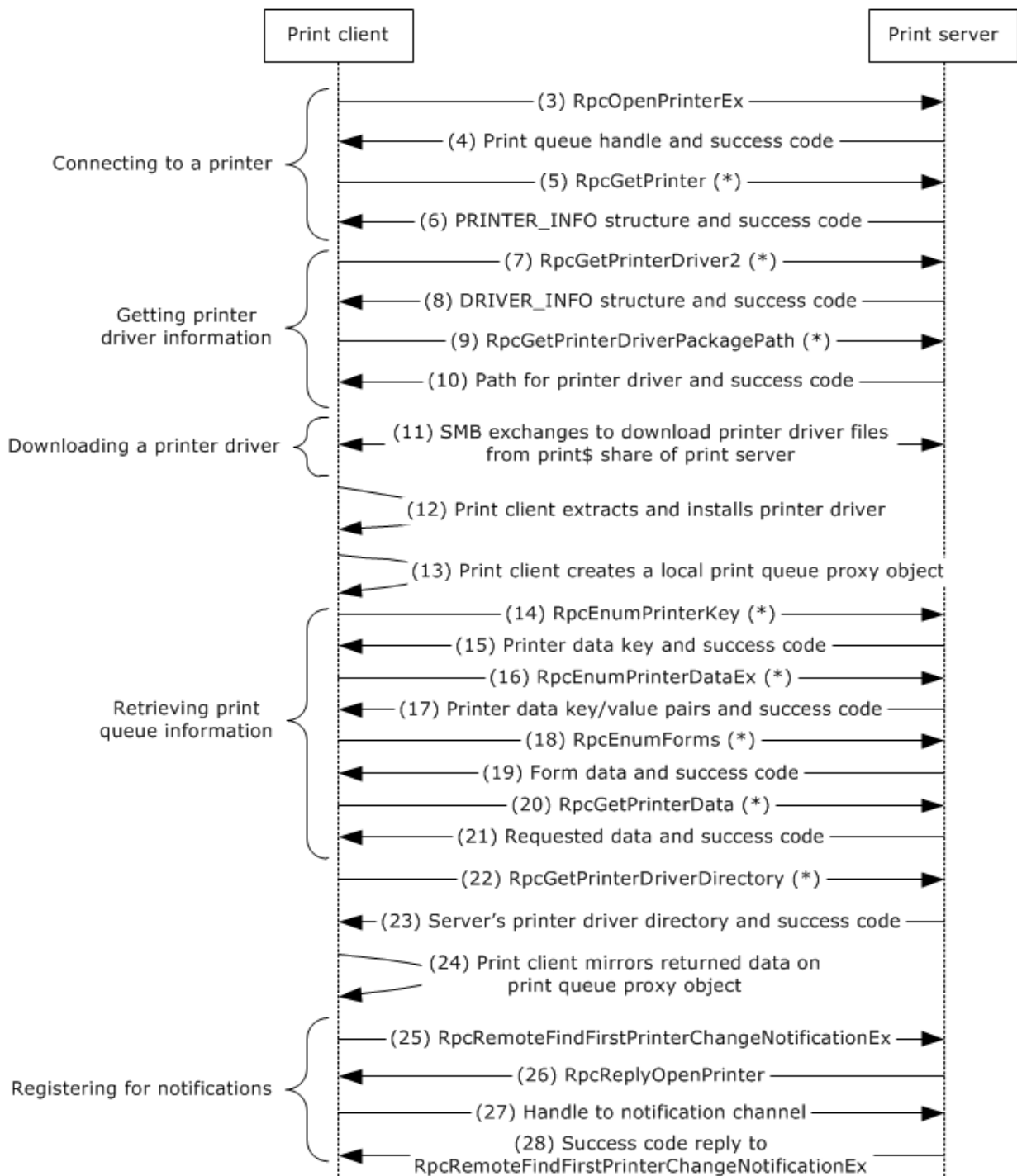
Figure 17: Print client connecting to a print queue, downloading a driver, and registering for notifications

The message groups in the preceding figure and their purpose are as follows:

Message group	Description	References
Connecting to a printer	Obtains the print server handle and information about the specified printer.	[MS-RPRN] (section 3.1.4.2)
Getting printer driver information	Obtains the driver information and package path of the specified printer driver.	[MS-RPRN] (section 3.1.4.4)
Downloading a printer driver	Downloads a printer driver using the SMB Version 1.0 protocol [MS-SMB].	[MS-SMB] (section 2.2) and [MS-FASOD] section 3.5
Retrieving print queue information	Obtains configuration data and form information for a print queue.	[MS-RPRN] (section 3.1.4.2) and [MS-RPRN] (section 3.1.4.5)
Registering for notifications	Creates a remote change notification object that monitors changes to printer objects and sends change notifications to the client.	[MS-RPRN] (section 3.1.4.10)

The following steps describe task 2 of this example:

3. The print client calls the **RpcOpenPrinterEx** method on the print server for the designated print queue.

4. The print server returns a handle to the print queue and a success code.

5. The print client calls the **RpcGetPrinter** (*) method on the print server for the print queue handle.

6. The print server returns **PRINTER_INFO** structure with details about the print queue and a success code.

7. The print client calls the **RpcGetPrinterDriver2** (*) method on the print server for the print queue handle.

8. The print server returns the requested **DRIVER_INFO** structure, which contains details about the print queue's printer driver, such as driver file name and driver version, and a success code.

9. The print client calls the **RpcGetPrinterDriverPackagePath** (*) method on the print server for the print queue handle.

10. The print server returns the path for the printer driver package cabinet file and a success code.

11. The print client downloads the printer driver using the SMB protocol family, utilizing the following File Services System operations, as defined in [MS-FASOD]: (a) Open File SMB ([MS-FASOD] section 3.5.13), (b) Perform File Operation SMB Query FS Information ([MS-FASOD] section 3.5.18), (c) Perform File Operation SMB Read ([MS-FASOD] section 3.5.19), (d) Perform File Operation SMB Close ([MS-FASOD] section 3.5.16).

12. The print client extracts the printer driver files contained in the downloaded data file and installs the printer driver.

13. The print client creates a connection to the print queue by creating a local print queue proxy object.

14. The print client calls the **RpcEnumPrinterKey** (*) method on the print server for the designated print queue handle.

15. The print server returns the requested printer data key and a success code.

16. The print client calls the **RpcEnumPrinterDataEx** (*) method on the print server for the designated print queue handle.

17. The print server returns the requested printer data key/value pairs and a success code.

18. The print client calls the **RpcEnumForms** (*) method on the print server for the designated print queue handle.

19. The print server returns the requested form data and a success code.

20. The print client calls the **RpcGetPrinterData** (*) method on the print server for the designated print queue handle to obtain values of specific keys.

21. The print server returns the requested data and a success code.

22. The print client calls the print server's **RpcGetPrinterDriverDirectory** method.<13>

23. The print server responds with the name of its printer driver directory and a success code.

24. The print client mirrors the returned data on the local print queue proxy object.

25. The print client calls the **RpcRemoteFindFirstPrinterChangeNotificationEx** method on the print server with a print queue handle to begin the process for registering to receive print status notifications.

26. The print server calls the **RpcReplyOpenPrinter** method on the print client to establish a notification channel.

27. The print client returns the **RpcReplyOpenPrinter** method call with a handle to the notification channel.

28. The print server returns the **RpcRemoteFindFirstPrinterChangeNotificationEx** method call with a success code.<14>

Task 3: Submitting a print job and receiving notifications

The following diagram illustrates the print client submitting a print job to the print queue and receiving notifications.
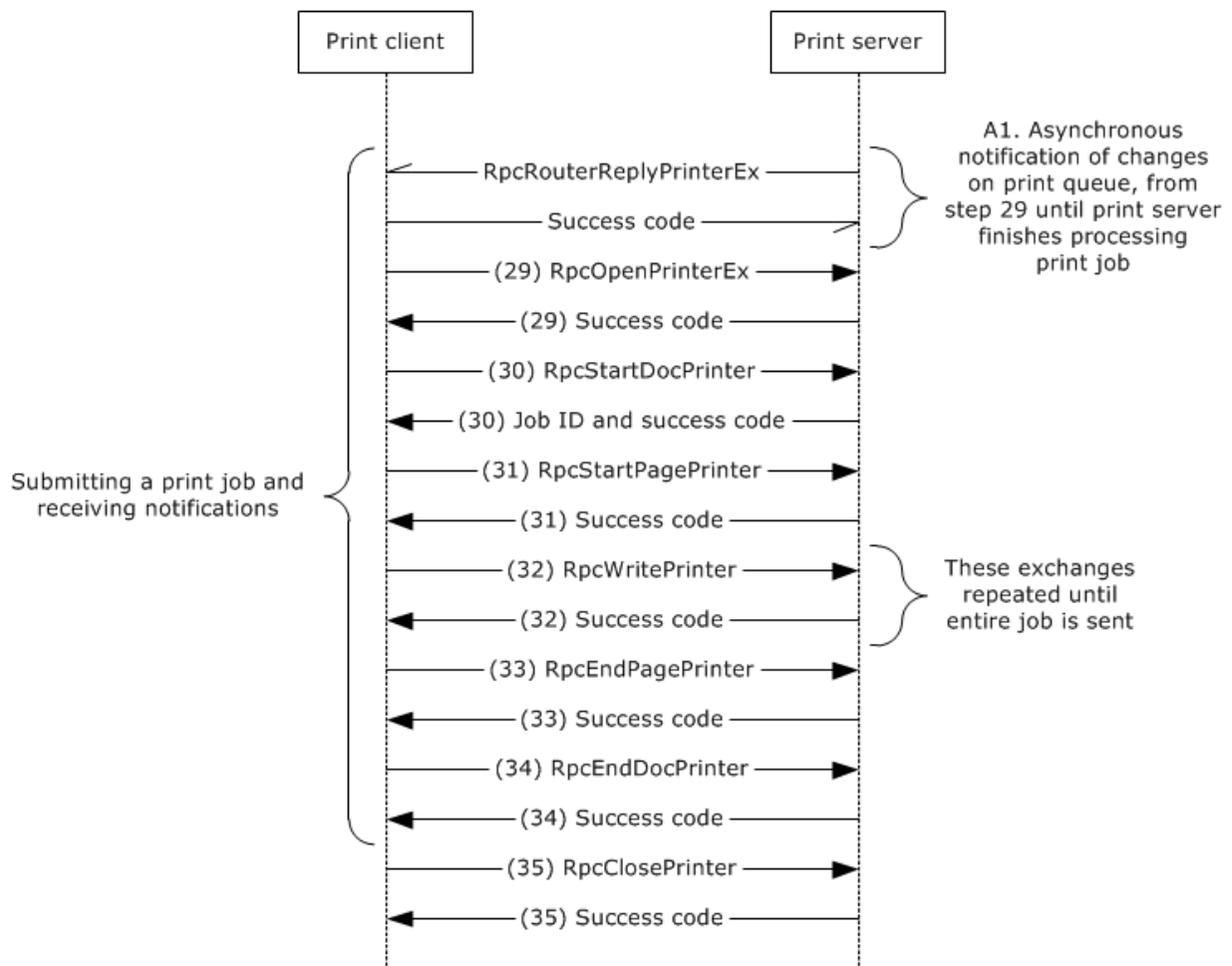
Figure 18: Print client submitting a print job to the print queue

The message group labeled "Submitting a print job and receiving notifications" in the preceding figure shows how the client receives asynchronous notification of changes on the print queue and how the client submits a print job, as described in [MS-RPRN] sections 3.1.4.9.1, 3.1.4.9.2, 3.1.4.9.3, 3.1.4.9.4, and 3.1.4.9.7.

The following steps describe task 3 of this example:

Asynchronously (on a separate thread), during each of the steps of task 3 and until the print server is done processing the print job, the print server calls the **RpcRouterReplyPrinterEx** method to notify the print client of changes on the print queue, such as job processing progress, status changes, or other changes. The print client responds to each notification with a success code.

29. The print client calls **RpcOpenPrinterEx** method on the print server to open a handle to the print queue for submitting a print job and the print server returns a success code.

30. The print client calls **RpcStartDocPrinter** method on the print server and the print server returns a job ID and a success code.

31. The print client calls the **RpcStartPagePrinter** method on the print server and the print server responds with a success code.

32. The print client repeatedly calls the **RpcWritePrinter** method on the print server, sending successive portions of the print job. The print server responds to each call with a success code. This step is repeated until all the print data has been sent.

33. The print client calls the **RpcEndPagePrinter** method on the print server and the print server responds with a success code.

34. When all print data has been sent, the print client calls the **RpcEndDocPrinter** method on the print server and the print server responds with a success code.

35. The print client calls the **RpcClosePrinter** method on the print server and the print server returns a success code. (This step will only occur if the print client does not proceed to the next portion of this example.)

Task 4: Unregistering for notifications

The following diagram illustrates the print client unregistering for notifications.
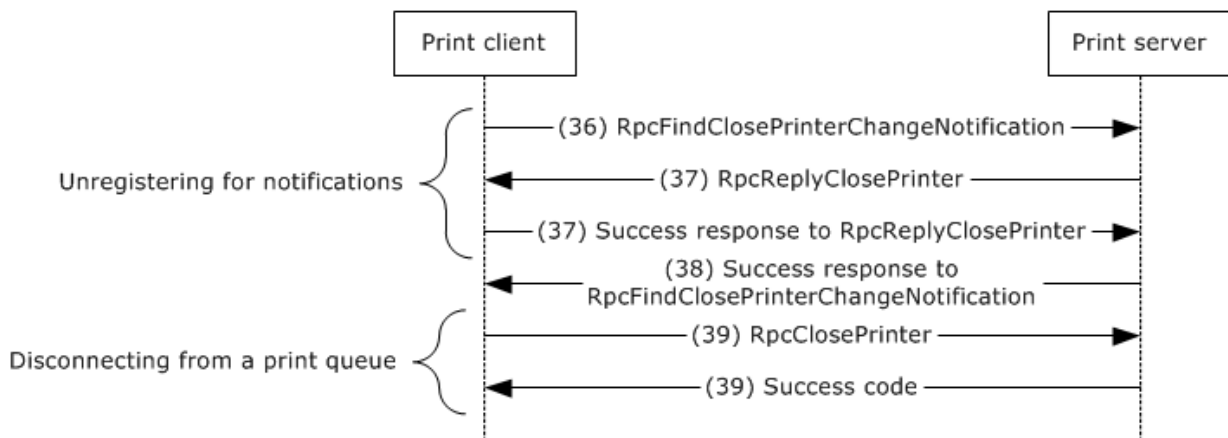
Figure 19: Print client unregistering for notifications

Message group	Description	References
Unregistering for notifications	The print client unregisters for notifications of changes to the print queue.	**[MS-RPRN] (section 3.1.4.10.2)**
Disconnecting from a print queue	The print client closes the handle to the printer object.	**[MS-RPRN] (section 3.1.4.2.9)**

The following steps describe task 4 of this example:

36. When a print client has no further use for notifications (for example, the user closes the print queue dialog user interface), the print client calls the **RpcFindClosePrinterChangeNotification** method on the print server.

37. The print server calls the **RpcReplyClosePrinter** method on the print client and the print client returns a success code.

38. The print server responds to the **RpcFindClosePrinterChangeNotification** method with a success code.

39. The print client calls the **RpcClosePrinter** method on the print server by using the handle to the print queue specified during registration and the print server returns a success code.

It is common for clients to listen for notifications from more than one thread, so the sequence of steps for registration for notifications, from task 2, and unregistration, from task 3, can be observed on the wire multiple times.

3.2 Example 2: Discovering and Utilizing a Print Queue in a Workgroup

This example demonstrates the use cases described in section 2.5.3.4 and section 2.5.3.7.1.

In this example, a computer located in a workgroup locates a shared print queue on another computer in the workgroup and sends a print job to the shared print queue.

In this example, there is no dedicated print server, no domain has been established, and no Directory System is available. The clients in the workgroup are distinguished by their roles, already established, as follows:

- Print server: One client in the workgroup is established as a print server and has already announced the shared print queues it hosts to all other print servers in the workgroup (if there are any), using the CIFS protocol [MS-BRWS]. The printers are shared by using the Print System Remote Protocol [MS-RPRN].<15>

- Print client: One client is distinguished as a print client in the workgroup when directed by the user to send a print job to a print server. This print client can behave in other roles, such as print server or print browser server, but for this example the client will behave in the role of a print client only.

- Print browser server: One client in the workgroup is established as the print browser server, which enumerates available print servers and print queues to print clients. The print browser server has already announced this role to all the clients in the workgroup using the protocol described in [MS-BRWS]. Print clients locate the print browser server using the CIFS and RAP protocols through a **NetServerEnum** method call in which the **Level** field is set to 101 and the **Flags** field to SV_TYPE_PRINTQ_SERVER. In order to act in the role of the print browser server, the print client also plays the role of a print server with at least one shared print queue available.

Only one print server in the workgroup behaves as the active print browser server at any given time. Other print servers (if any) behave as backup print browser servers, and one of them will become the new active print browser server if the current active print browser server becomes unavailable. The print browser server and the backup print browser servers exchange information about shared print queues in the workgroup by calling the **RpcAddPrinter** method [MS-RPRN] (section 3.1.4.2.3) with the **PRINTER_INFO_*** structures on each other.

Typically, the first workgroup client that takes on the role of a print server sharing a print queue also becomes the print browser server for the workgroup.

Prerequisites

This example has the following prerequisites:

- The general requirements as set forth in Assumptions and Preconditions, section 2.4, are met.

- Participating clients are configured to be members of the same workgroup.

- The \pipe\spoolss endpoint can be accessed anonymously.

- All participating clients support the protocol described in [MS-BRWS].

- One of the clients has at least one shared print queue, so that at least one client will be of type **SV_TYPE_PRINTQ_SERVER**, which is a requirement for this example.

- Either the print client or print server supports the Print System Remote Protocol [MS-RPRN], but not the Print System Asynchronous Remote Protocol [MS-PAR].<16> This precondition was chosen to illustrate the use of Print System Remote Protocol.<17>

- The print client is configured to not use Windows Update to obtain the printer driver. This precondition is present to illustrate printer driver download from the print server.

- The printer driver is available from the print server's print$ share.

Initial System State

This example applies in the following initial system state.

- The print client does not have the printer driver for the print queue.

- The print client does not have the printer driver package for the printer driver in its local driver store. This state is present to illustrate printer driver download from the print server.

Final System State

- The print client obtains the list of available print servers and print queues.

- The print client has downloaded a printer driver for the print queue.

This example is divided into four tasks. The steps for each task are shown in the following sections.

1. Locating a print queue in a workgroup

2. Establishing a connection and registering for notifications

3. Submitting a print job and receiving notifications

4. Unregistering for notifications

Task 1: Locating a print queue in a workgroup

The following diagram illustrates the first part of this example, in which a print client locates a print queue in a workgroup and establishes a connection to the print queue. The sequence is described in the steps which follow the diagram.

Release: Thursday, February 13, 2014
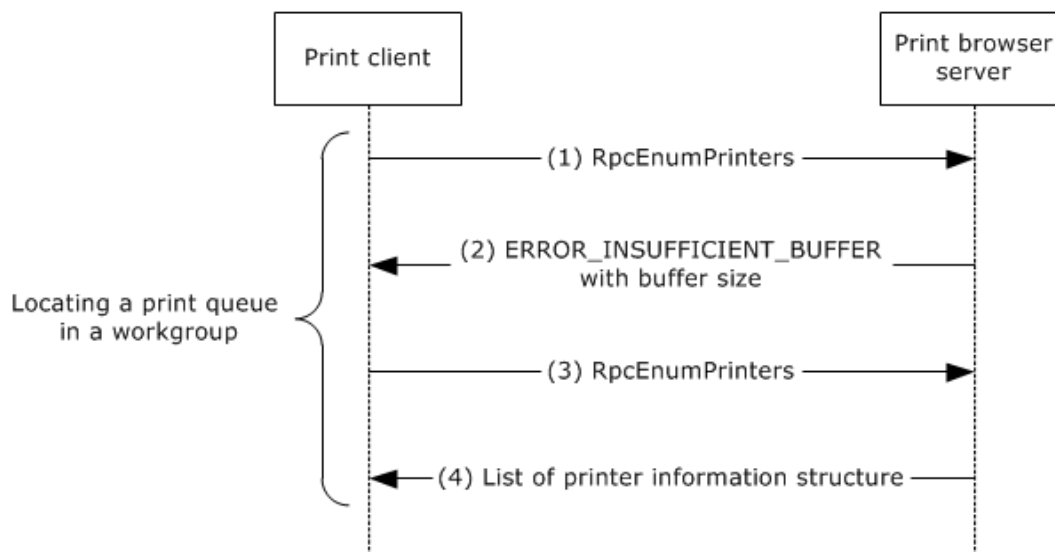
Figure 20: Print client locating a print queue in a workgroup

The message group labeled "Locating a print queue in a workgroup" shows how the client enumerates the printers in the workgroup, as described in **[MS-RPRN] (section 3.1.4.2.1)**.

The following steps describe task 1 of this example.

1. The print client calls the **RpcEnumPrinters** method on the print browser server using the protocol described in [MS-RPRN], requesting whatever level of printer information structure the client requires as described in **[MS-RPRN]** (section 3.1.4.2.1).

2. The server returns ERROR_INSUFFICIENT_BUFFER and sets the *countBytesNeeded* parameter to the size needed to store the requested printer information structures for all shared print queues.

3. The client allocates an amount of memory equal to *countBytesNeeded*. The client calls **RpcEnumPrinters** and passes the allocated memory as a parameter.

4. The server stores a printer information structure for each shared print queue in the output buffer, stores the number of print information structures in the *printersFound* parameter, and returns ERROR_SUCCESS.

   If the number of shared print queues on the server has increased between the first and second call to **RpcEnumPrinters**, the server returns ERROR_INSUFFICIENT_BUFFER from the second call as well. In that case, the server will update the *countBytesNeeded* parameter so that the client can reallocate the buffer and repeat the call to **RpcEnumPrinters**.

   Task 2: Establishing a connection and registering for notifications

   The following diagram illustrates the second part of this example, in which a print client establishes a connection to a print queue and registers for notifications.
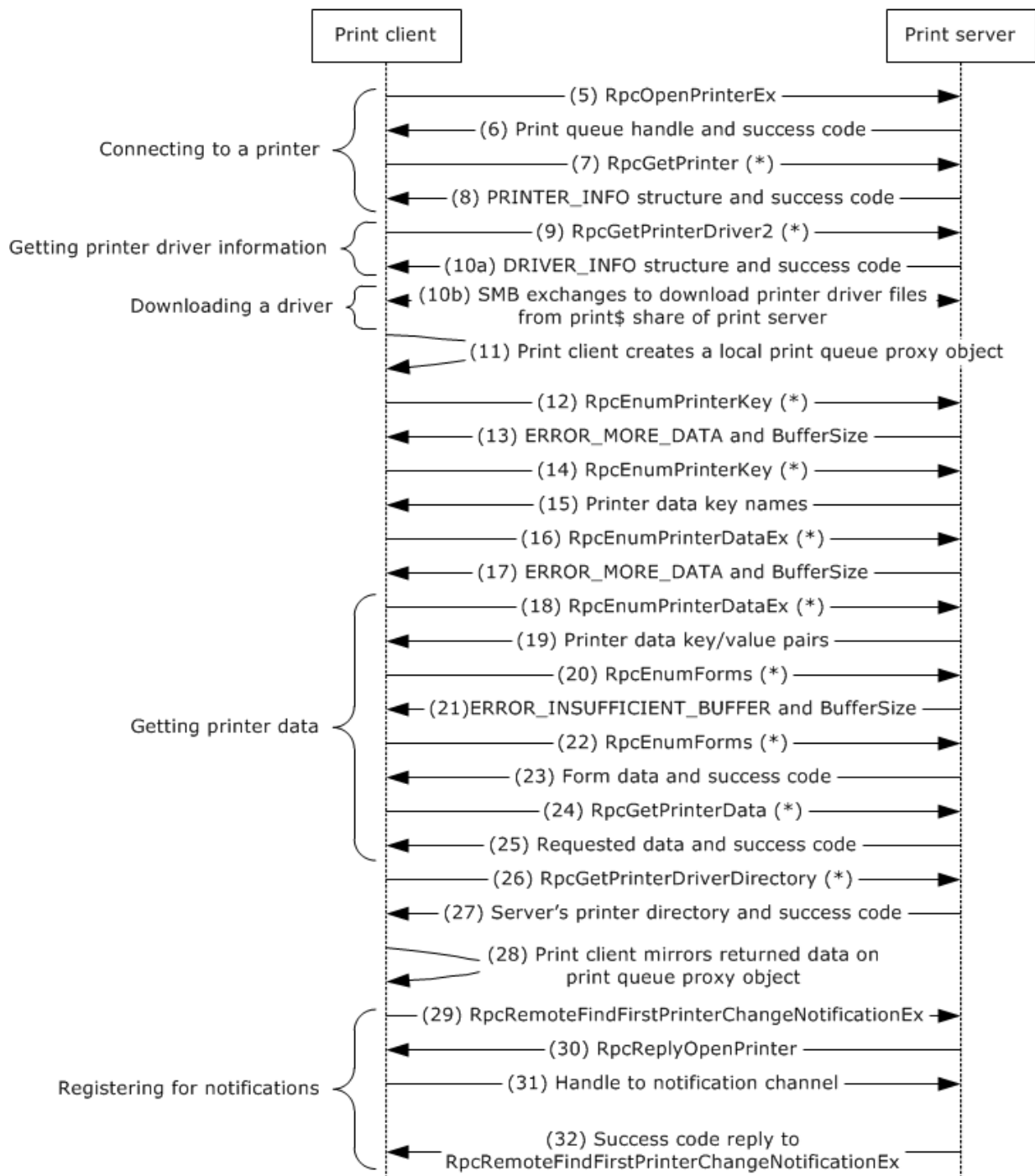
Figure 21: Print client establishing a connection and registering for notifications

Message group	Description	References
Connecting to a printer	The client obtains the print server handle and information about the specified printer.	[MS-RPRN] (section 3.1.4.2)

Message group	Description	References
Getting printer driver information	The client obtains information about the device driver for the specified printer.	[MS-RPRN] (section 3.1.4.4)
Downloading a driver	The client downloads a printer driver using the protocol described in [MS-SMB].	[MS-SMB] (section 2.2) and [MS-FASOD] section 3.5.
Getting printer data	The client obtains configuration data and form information for a print queue.	[MS-RPRN] sections 3.1.4.2 and 3.1.4.5
Registering for notifications	The client creates a remote change notification object that monitors changes to printer objects and sends notifications of those changes to the client using the **RpcRouterReplyPrinter** method.	[MS-RPRN] (section 3.1.4.10)

The following steps describe task 2 of this example:

It is common for clients to listen for notifications from more than one thread, so the following sequence of steps for registration for notifications can be observed on the wire multiple times.

5. The print client calls the **RpcOpenPrinterEx** method on the print server for the designated print queue.

6. The print server returns a handle to the print queue and a success code.

7. The print client calls the **RpcGetPrinter** (*) method on the print server for the print queue handle.

8. The print server returns **PRINTER_INFO** structure with details about the print queue and a success code.

9. The print client calls the **RpcGetPrinterDriver2** (*) method on the print server for the print queue handle.

10. This step consists of the following two parts:

    1. The print server returns the requested **DRIVER_INFO** structure, which contains details about the print queue's printer driver, such as the driver's file name and version, and a success code.

    2. The print client downloads the printer driver files from the print$ share of the print server using the SMB protocol family (the print client has no control over whether the SMB Version 1.0 or SMB Version 2.0 protocol is used).

11. The print client creates a connection to the print queue by creating a local print queue proxy object.

12. The print client calls the **RpcEnumPrinterKey** (*) method on the print server for the designated print queue handle.

13. The print server returns ERROR_MORE_DATA and sets the buffer size needed for the subkey.

14. The client allocates a buffer with the size as returned by the server. The client calls **RpcEnumPrinterKey**, passing the buffer as a parameter.

15. The print server returns the requested printer data subkey of the specified key and a success code.

16. The print client calls the **RpcEnumPrinterDataEx** (*) method on the print server for the designated print queue handle.

17. The print server returns ERROR_MORE_DATA and sets the buffer size needed for enumerating all value names and data for the specified printer and key.

18. The client allocates a buffer with the size as returned by the server. The client calls **RpcEnumPrinterDataEx** and passes the allocated memory as a parameter.

19. The print server returns the requested printer data key/value pairs and a success code.

20. The print client calls the **RpcEnumForms** (*) method on the print server for the designated print queue handle.

21. The print server returns ERROR_INSUFFICIENT_BUFFER and sets the buffer size needed for the forms that the printer supports.

22. The client allocates a buffer with the size as returned by the server. The client calls **RpcEnumForms** and passes the allocated memory as a parameter.

23. The print server returns the requested form data and a success code.

24. The print client calls the **RpcGetPrinterData** (*) method on the print server for the designated print queue handle to obtain values of specific keys.

25. The print server returns the requested data and a success code.

26. The print client calls the print server's **RpcGetPrinterDriverDirectory** method.<18>

27. The print server responds with the name of its printer driver directory and a success code.

28. The print client mirrors the returned data on the local print queue proxy object.

29. The print client calls the **RpcRemoteFindFirstPrinterChangeNotificationEx** method on the print server with a print queue handle to begin the process for registering to receive print status notifications.

30. The print server calls the **RpcReplyOpenPrinter** method on the print client to establish a notification channel.

31. The print client returns the **RpcReplyOpenPrinter** method call with a handle to the notification channel.

32. The print server returns the **RpcRemoteFindFirstPrinterChangeNotificationEx** method call with a success code.<19>

Task 3: Submitting a print job and receiving notifications

The following diagram illustrates the print client submitting a print job to the print queue.
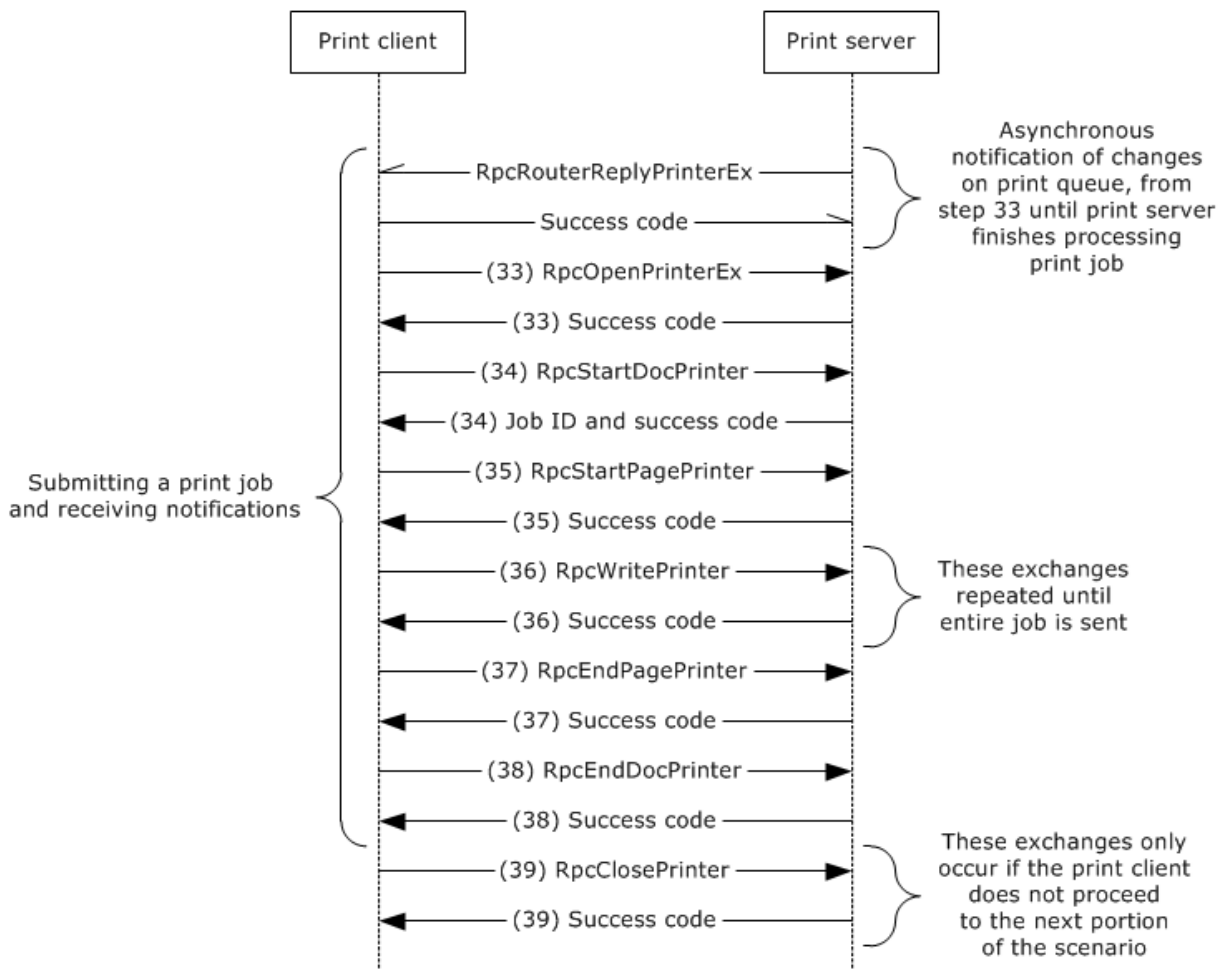
Figure 22: Print client submitting a print job to the print queue

The message group labeled "Submitting a print job and receiving notifications" in the preceding figure shows how the client receives asynchronous notification of changes on a print queue and how it submits a print job, as described in [MS-RPRN] sections 3.1.4.9.1, 3.1.4.9.2, 3.1.4.9.3, 3.1.4.9.4, and 3.1.4.9.7.

The following steps describe task 3 of this example:

Asynchronously, and until the print server is done processing the print job, the print server calls the **RpcRouterReplyPrinterEx** method on the print client to notify the client of changes on the print queue, such as job processing progress, status changes, or other changes.

33. The print client calls the **RpcOpenPrinterEx** method on the print server to open a handle to the print queue for submitting a print job and the print server returns a success code.

34. The print client calls the **RpcStartDocPrinter** method on the print server and the print server returns a job ID and a success code.

35. The print client calls the **RpcStartPagePrinter** method on the print server and the print server responds with a success code.

36. The print client repeatedly calls the **RpcWritePrinter** method on the print server, sending successive portions of the print job. The print server responds to each call with a success code. This step is repeated until all the print data has been sent.

37. The print client calls the **RpcEndPagePrinter** method on the print server and the print server responds with a success code.

38. When all print data has been sent, the print client calls the **RpcEndDocPrinter** method on the print server and the print server responds with a success code.

39. The print client calls the **RpcClosePrinter** method on the print server and the print server returns a success code. (This step will only occur if the print client does not proceed to the next portion of this example.)

Task 4: Unregistering for notifications

The following diagram illustrates the print client unregistering for notifications.
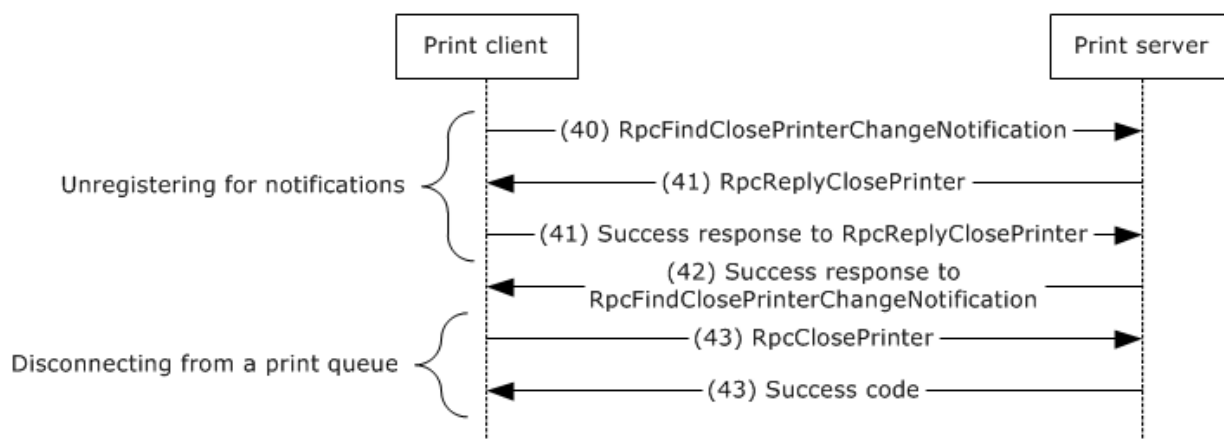
Figure 23: Print client unregistering for notifications

Message group	Description	References
Unregistering for notifications	The client unregisters for notification of changes on the print queue.	**[MS-RPRN] (section 3.1.4.10.2)**
Disconnecting from a print queue	The client closes its handle to the printer object.	**[MS-RPRN] (section 3.1.4.2.9)**

The following steps describe task 4 of this example:

It is common for clients to listen for notifications from more than one thread, so the sequence of steps for unregistration for notifications can be observed on the wire multiple times.

40. When a print client is no longer consuming notifications (for example, the user closes the print queue dialog user interface), the print client calls the **RpcFindClosePrinterChangeNotification** method on the print server.

41. The print server calls the **RpcReplyClosePrinter** method on the print client and the print client returns a success code.

42. The print server responds to the **RpcFindClosePrinterChangeNotification** method with a success code.

43. The print client calls the **RpcClosePrinter** method on the print server using the handle to the print queue specified during registration and the print server returns a success code.

3.3   Example 3: Receiving Unidirectional IHV-Defined Notifications

This example demonstrates the use cases described in section 2.5.3.7.1 Extension (b), wherein the print client receives IHV-defined notifications from a manual duplex printer using the Print System Asynchronous Notification Protocol [MS-PAN] after a print queue has been located in a domain, connection has been established, a print job has been submitted by using the Print System Remote Protocol [MS-RPRN] and the Print System Asynchronous Remote Protocol [MS-PAR], and unidirectional IHV-defined communication has been established between the print client and print server by using the Print System Asynchronous Notification Protocol [MS-PAN].

Prerequisites

This example has the following prerequisites:

- The print client and print server support the protocols described in [MS-RPRN], [MS-PAR], and [MS-PAN].<20>

- There is an IHV-defined print processor installed on the print server that serves as the Print System Asynchronous Notification Protocol [MS-PAN] notification source and implements a Manual Duplex operation, which prompts the user to walk up to the printer and reinsert printed pages.

- The print queue has been provisioned and is located on a print server in a domain.

- The print server has published a list of the shared print queues to the Directory System.

- The print client has located the print queue in the domain using the LDAP protocol [RFC4511].

- The printer driver is available from the print server.

- The print client has established a connection, downloaded a driver, registered for notifications, and submitted a print job, except that the asynchronous calls provided by the Print System Asynchronous Remote Protocol [MS-PAR] have been used instead of the corresponding the Print System Remote Protocol [MS-RPRN] calls that were used in Example 1.

Initial System State

There are no specific initial system state requirements for this example other than those implied by the prerequisites.

Final System State

The final system state is identical to the initial system state. This sequence of events leaves the system state unmodified.

Sequence

Asynchronous Communication: When the print client's notification listener thread receives a notification announcing a new print job, the print client starts another listener thread to register and listen for notifications. The notification process based on the Print System Asynchronous Notification

Protocol [MS-PAN] is illustrated in the following diagram, with further information in [MS-PAN] section 4.2.
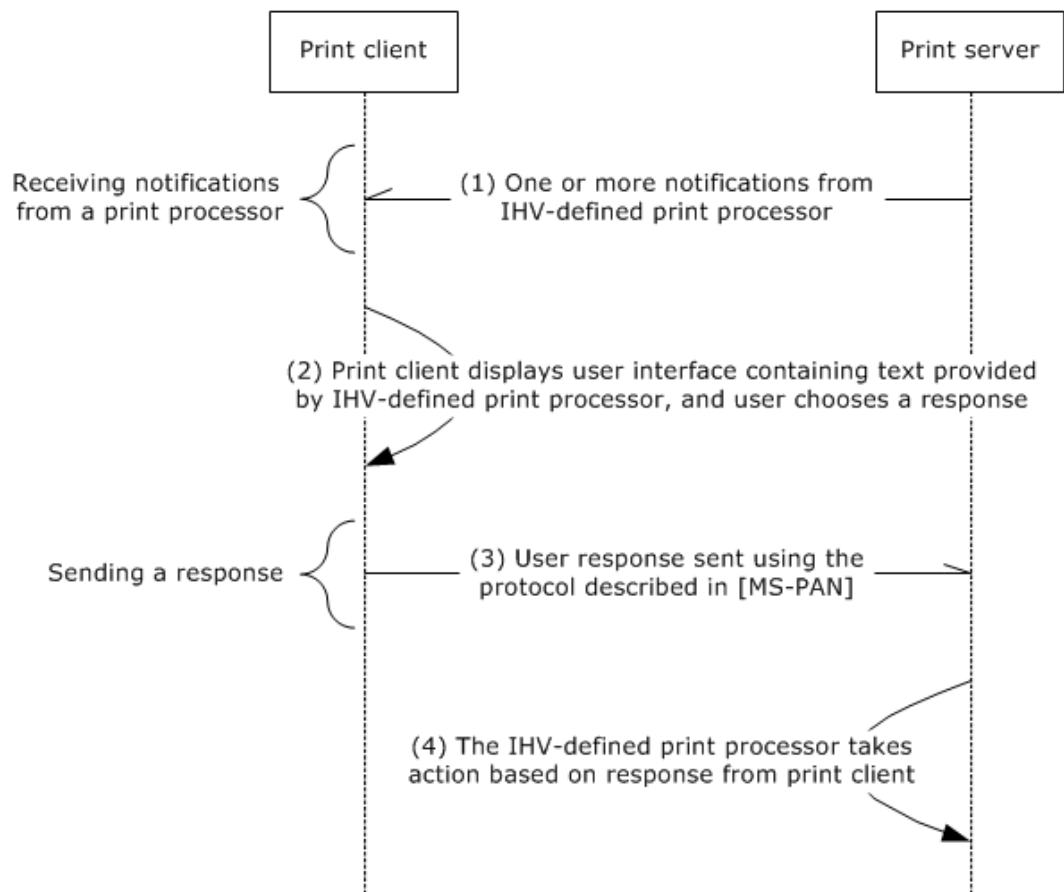
Figure 24: Print client receiving notifications

The particular messages within the message groups shown in the preceding figure are described in [MS-PAN] (section 4.2).

1. During processing of the print job on the print server, the IHV-defined print processor sends one or more notifications to the print client by using the Print System Asynchronous Remote Protocol [MS-PAR]. The IHV-defined print processor can send these notifications at arbitrary times.

2. When receiving the notification from the print server, the print client shows user interface to the user displaying custom text provided by the IHV-defined print processor. The user takes appropriate action and confirms the user interface.

3. The print client sends a response by using the Print System Asynchronous Remote Protocol [MS-PAR].

4. The IHV-defined print processor takes appropriate actions based on the response received from the print client.

It is common for print clients to listen for notifications from more than one thread, so the steps for registration for notifications and unregistration can be observed on the wire multiple times.

3.4   Example 4: Enumerating Print Jobs from All Users, Then Canceling Several Print Jobs

This example demonstrates the use case described in section 2.5.3.8.2.

An administrator with appropriate privileges can view and override print jobs submitted by other users. This example illustrates this administrative procedure, using only the Print System Remote Protocol [MS-RPRN].

Prerequisites

This example has the following prerequisites.

- The administrative client is being used by an administrator who has privileges for overriding the print operations of other users.

- Either the print client or print server supports the Print System Remote Protocol [MS-RPRN], but not the Print System Asynchronous Remote Protocol [MS-PAR]. This precondition was chosen to illustrate the use of the Print System Remote Protocol.<21>

Initial System State

This example has the following initial system state.

- Multiple users have submitted print jobs to the same print queue.

Sequence

The following diagram illustrates this example.
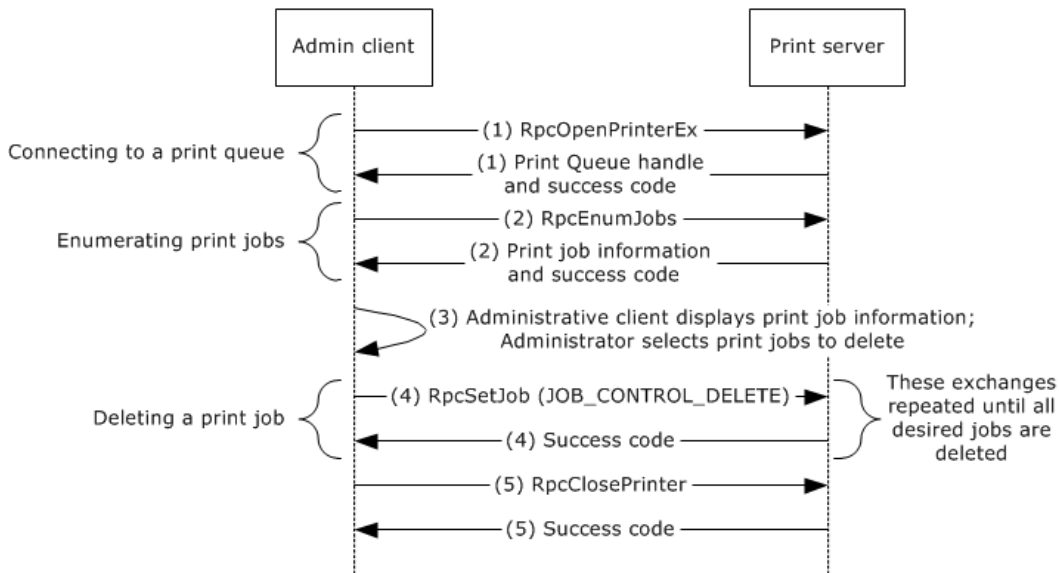
Figure 25: Administrative client enumerating and canceling print jobs on a print queue

Message group	Description	References
Connecting to a print queue	The client obtains a handle to the print server.	**[MS-RPRN] (section 3.1.4.2.14)**

Message group	Description	References
Enumerating print jobs	The client obtains a list of print jobs on the print server.	**[MS-RPRN] (section 3.1.4.3.3)**
Deleting a print job	The client deletes a print job from the print server.	**[MS-RPRN] (section 3.1.4.3.1)**

The following steps describe this sequence.

1. The administrative client calls the **RpcOpenPrinterEx** method on the print server to obtain the handle for the print queue. The print server returns the print queue handle and a success code.

2. The administrative client calls the **RpcEnumJobs** method on the print server using the print queue handle and the print server enumerates all the print jobs assigned to that print queue, and information about the print jobs (including names of the jobs, IDs of the jobs, and names of the users who submitted the jobs), and a success code.

3. The administrative client displays the information about the print jobs. The administrator selects the print jobs to cancel.

4. The administrative client repeatedly calls the **RpcSetJob** method (with the **JOB_CONTROL_DELETE** command) on the print server using the print queue handle, canceling one job at a time. The print server terminates each designated print job, deleting it from the print queue and responding to each method call with a success code.

5. The administrative client calls the **RpcClosePrinter** method on the print queue handle, and the print server responds with a success code.

Final State

The print queue in this example contains only print jobs that were not deleted by the administrator.

3.5   Example 5: Provisioning a Print Queue Using the Protocol Defined in [MS-RPRN]

This example demonstrates the use case described in section 2.5.3.1 and section 2.5.3.2 (variation (a) with extension (a)), wherein a print queue is provisioned from an administrative client using the Print System Remote Protocol [MS-RPRN] and then the same print queue is deleted from a different administrative client by using the Print System Asynchronous Remote Protocol [MS-PAR].

In this example, two different administrative clients are involved in managing print queues: the first client provisions a print queue and the second client subsequently deletes the same print queue. The first client uses the Print System Remote Protocol [MS-RPRN]; the second client uses the Print System Asynchronous Remote Protocol [MS-PAR]. Although both management functions can be accomplished by either protocol, this example demonstrates how the parallel functionality can be accomplished using different implementations or different versions of administrative clients.

Windows administrative clients that are using the Print System Asynchronous Remote Protocol [MS-PAR] to manage print queues will first use the Print System Remote Protocol [MS-RPRN] to ensure that the Print System Asynchronous Remote Protocol is supported by the print server and that the print server is available.

Prerequisites

This example has the following prerequisites.

- Administrative clients and print servers are located within a domain configuration.

- Administrative clients and print servers have access to the Active Directory System provided by the domain.

- Administrator of section A has physically connected a printer to a print server or known network location.

- Administrator of section A knows the location from which to obtain a printer driver for the newly attached printer.

- The administrative client used to provision the print queue supports the Print System Remote Protocol [MS-RPRN], but not the Print System Asynchronous Remote Protocol [MS-PAR].<22>

- The administrative client used to delete the print queue and print server supports the Print System Remote Protocol [MS-RPRN] and the Print System Asynchronous Remote Protocol [MS-PAR].

Initial System State

This example has the following initial system state.

- The print queue is not added to the print server.

- The print queue is not published to the LDAP server.

Final System State

- The final system state is identical to the initial system state.

This example is divided into two tasks.

1. Provisioning a print queue through the Print System Remote Protocol

2. Deleting that print queue through the Print System Asynchronous Remote Protocol

Task 1: Provisioning a print queue through the Print System Remote Protocol

The following diagram illustrates the first task of this example.
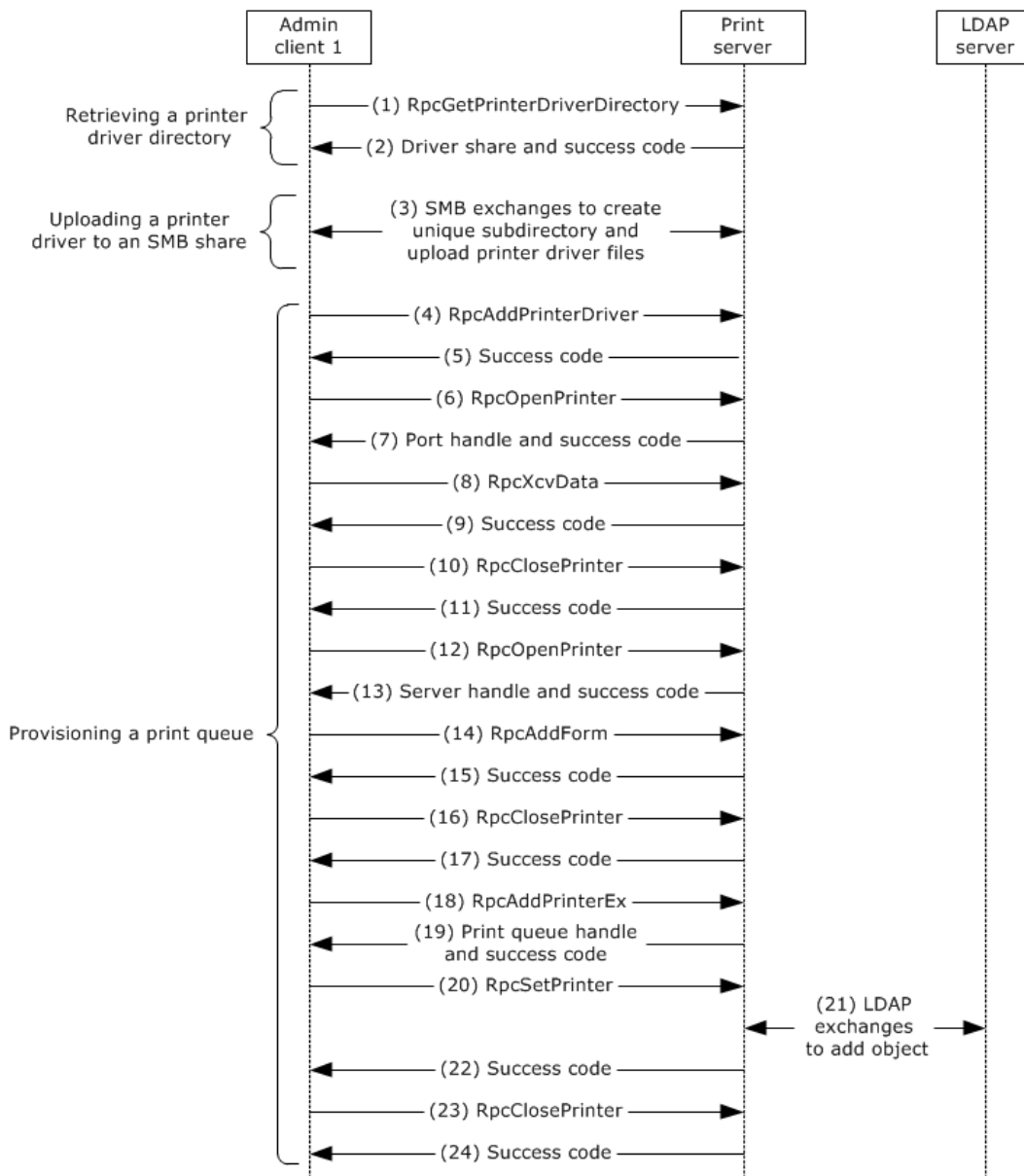
**Figure 26: Administrative client provisioning a print queue by using the Print System Remote Protocol**

Message group	Description	Reference
Retrieving a printer driver directory	The administrative client retrieves the directory where printer drivers are stored.	**[MS-RPRN] (section 3.1.4.4.4)**
Uploading a printer driver to an SMB share	The administrative client uses the SMB protocol to upload a printer driver to the printer driver directory.	[MS-SMB] (section 2.2) and [MS-FASOD] section 3.5
Provisioning a print	The administrative client provisions the print queue,	[MS-RPRN] (section

Message group	Description	Reference
queue	which involves adding a printer, adding a form, and setting the print queue's printer property.	3.1.4)

The following steps describe task 1 of this example.

1. Administrative client 1 uses the Print System Remote Protocol to call the **RpcGetPrinterDriverDirectory** method on the print server to retrieve the printer driver share (see [MS-RPRN] section 3.1.4.4.4).

2. The print server returns the requested path for the driver share and a success code.

3. Administrative client 1 uses the protocol family described in [MS-SMB] to create a unique subdirectory in the print share of the print server and uploads the files comprising the printer driver to that location.

4. Administrative client 1 calls the **RpcAddPrinterDriver** method on the print server, specifying the unique directory created in Step 3.

5. The print server returns a success code.

6. Administrative client 1 calls the **RpcOpenPrinter** method on the print server to open a port handle for the port monitor.

7. The print server returns a port handle and a success code.

8. The administrative user initiates an **Add Port** action in the administrative tool and selects to add a TCP/IP port to the server. In response, the administrative client 1 calls the **RpcXcvData** method on the print server, specifying an **Add Port** action.

9. The print server adds a port and returns a success code.

10. Administrative client 1 calls the **RpcClosePrinter** method on the print server to close the port handle.

11. The print server returns a success code.

12. Administrative client 1 calls the **RpcOpenPrinter** method on the print server to open a server handle to the print server.

13. The print server returns a server handle and a success code.

14. The administrative user initiates an **Add Form** action in the administrative tool and adds a form. In response, the administrative client 1 calls the **RpcAddForm** method on the print server to add a form.

15. The print server adds the form and returns a success code.

16. Administrative client 1 calls the **RpcClosePrinter** method on the print server to close the server handle.

17. The print server returns a success code.

18. Administrative client 1 calls the **RpcAddPrinterEx** method on the print server, using the **PRINTER_INFO_2** structure, the **DEVMODE_CONTAINER** structure, and the **SECURITY_CONTAINER** structure.

19. The print server returns a handle to the new print queue and a success code.

20. Administrative client 1 calls the **RpcSetPrinter** method on the print server, using the **DEVMODE_CONTAINER** structure, the **SECURITY_CONTAINER** structure, and the **PRINTER_INFO_7** structure (specifying that the print queue be published in the Active Directory System).

21. The print server uses the LDAP protocol [RFC4511] to add the print queue as a new object in the Active Directory System, indicating the location of the print queue.

22. The print server returns a success code to the print client.

23. Administrative client 1 calls the **RpcClosePrinter** method on the print server.

24. The print server returns a success code to the Administrative Client.

Task 2: Deleting that print queue through the Print System Asynchronous Remote Protocol

The following diagram illustrates the second task of this example.
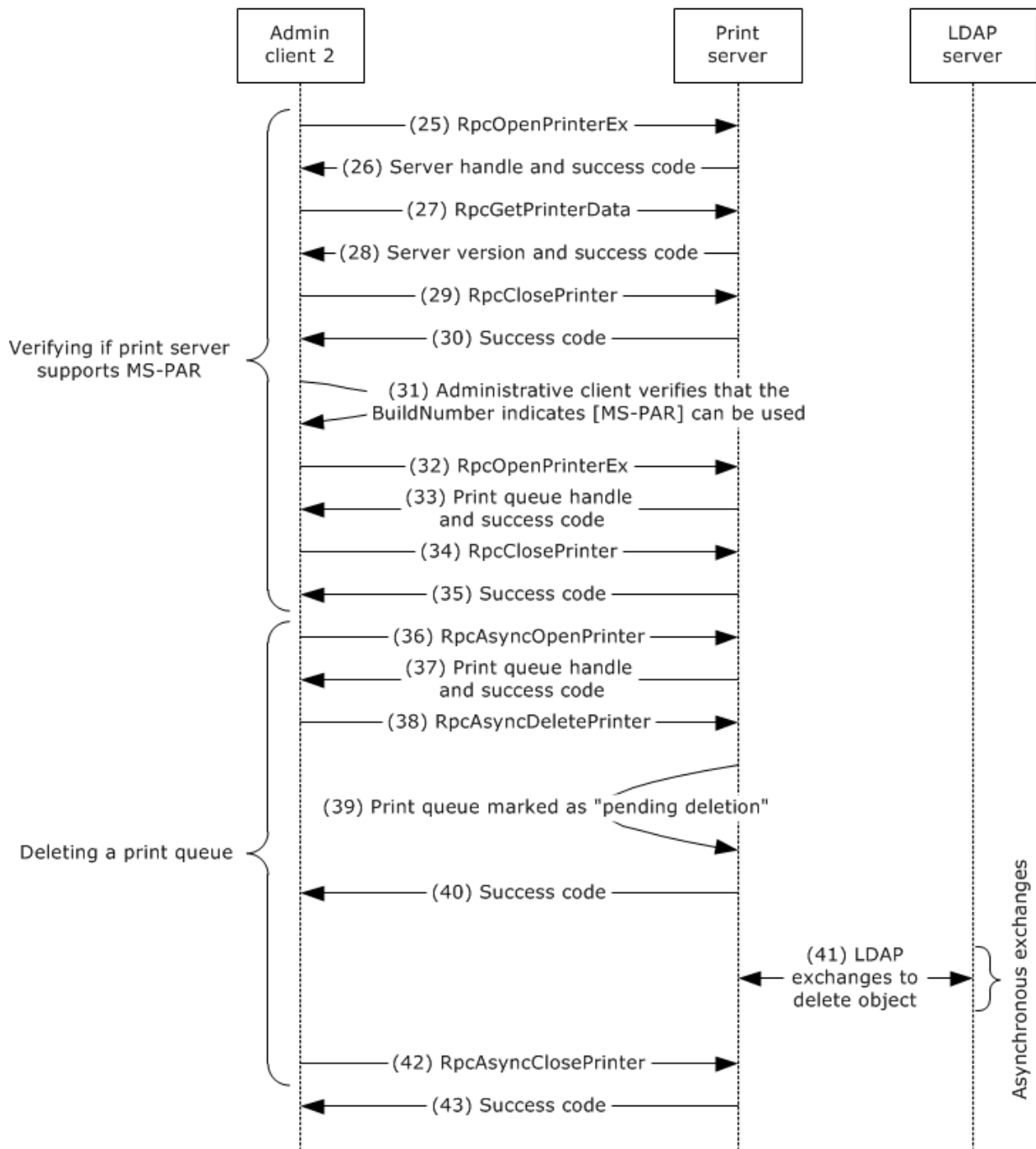
Figure 27: Administrative client deleting a print queue by using the Print System Asynchronous Remote Protocol

Message group	Description	References
Verifying a	The administrative client uses the Print System Remote	[MS-RPRN]

Message group	Description	References
print server	Protocol to verify that the print server supports the Print System Asynchronous Remote Protocol and is available.	(section 2.2)
Deleting a print queue	The administrative client uses the Print System Asynchronous Remote Protocol to delete the print queue.	[MS-PAR] (section 3.1.4.1)

The following steps describe task 2 of this example.

25. Administrative client 2 uses the Print System Remote Protocol [MS-RPRN] to call the **RpcOpenPrinterEx** method on the print server to obtain a print server handle.

26. The print server returns the server handle and success code.

27. Administrative client 2 uses the Print System Remote Protocol [MS-RPRN] to call the **RpcGetPrinterData** method on the print server, specifying the server handle and the OSVersion key.

28. The print server returns the server version, which contains the BuildNumber, and a success code.

29. Administrative client 2 uses the Print System Remote Protocol [MS-RPRN] to call the **RpcClosePrinter** method on the print server for the server handle.

30. The print server returns a success code.

31. The Administrative client 2 verifies that the BuildNumber (OSVersion value) is 3791 or greater, indicating that the Print System Asynchronous Remote Protocol [MS-PAR] can be used for managing print queues.

32. Administrative client 2 uses the Print System Remote Protocol [MS-RPRN] to call the **RpcOpenPrinterEx** method on the print server for the print queue to obtain the print queue handle. This step and the next two are used to verify that the print server is still available.

33. The print server returns the print queue handle and a success code.

34. Administrative client 2 uses the Print System Remote Protocol [MS-RPRN] to call the **RpcClosePrinter** method on the print server for the print queue handle used for verification that the print server is still available.

35. The print server returns a success code.

36. Administrative client 2 uses the Print System Asynchronous Remote Protocol [MS-PAR] to call the **RpcAsyncOpenPrinter** method on the print server for the specified print queue.

37. The print server returns the print queue handle and a success code.

38. Administrative client 2 uses the Print System Asynchronous Remote Protocol [MS-PAR] to call the **RpcAsyncDeletePrinter** method on the print server for the print queue handle.

39. The print server marks the specified print queue as "pending deletion".

40. The print server returns a success code.

41. Following the previous step, on a separate thread of execution, the print server does the following:

1. Uses LDAP to remove the published print queue from the LDAP server.

2. Waits until the print queue marked "pending deletion" is no longer in use by any other print client or active print job, and then deletes the print queue from the print server.

42. Administrative client 2 uses the Print System Asynchronous Remote Protocol [MS-PAR] to call the **RpcAsyncClosePrinter** method on the print server for the print queue handle.

43. The print server returns a success code.

3.6 Example 6: Sending a Print Job to an SMB Share

This example demonstrates the use cases described in sections 2.5.3.7.3 and 2.5.3.8.3.

This example involves the SMB protocol family [MS-SMB] and the RAP protocol [MS-RAP], supporting interoperability with non-Windows platforms.<23> This example also applies to the rare case when a Windows print client installs a local print queue with a local port specifying the UNC path of a shared print queue on a print server.<24>

In this example, when a user performs commands from the command line, the user is interacting with the SMB Server Service installed on the print server. A user sends a print job to an SMB share by using the command **copy** /b file \\server\printer at the command line. The print client subsequently uses the SMB protocol family to copy the file to the specified SMB print share of the print server. The SMB/RAP Print Redirector module (part of the SMB protocol family Server Service) on the print server translates SMB calls from the print client to local calls to the print subsystem of the print server.

After sending a print job to the SMB share of the print server, the user can obtain a list of the active jobs on the specified print queue by using the command **net print** \\server\printer<25> at the command line. The print client uses RAP to obtain the list of active jobs. The SMB/RAP Print Redirector module on the print server translates the RAP calls from the print client to local calls to the print subsystem of the print server.

Prerequisites

This example has the following initial system state and prerequisites.

▪ The requirements in the Assumptions and Preconditions (section 2.4) are met.

▪ The SMB Server Service is installed and active on the print server.

Initial System State

This example has the following initial system state.

▪ The print job is ready to be sent to the specified SMB print share.

Sequence

The following diagram illustrates this example.
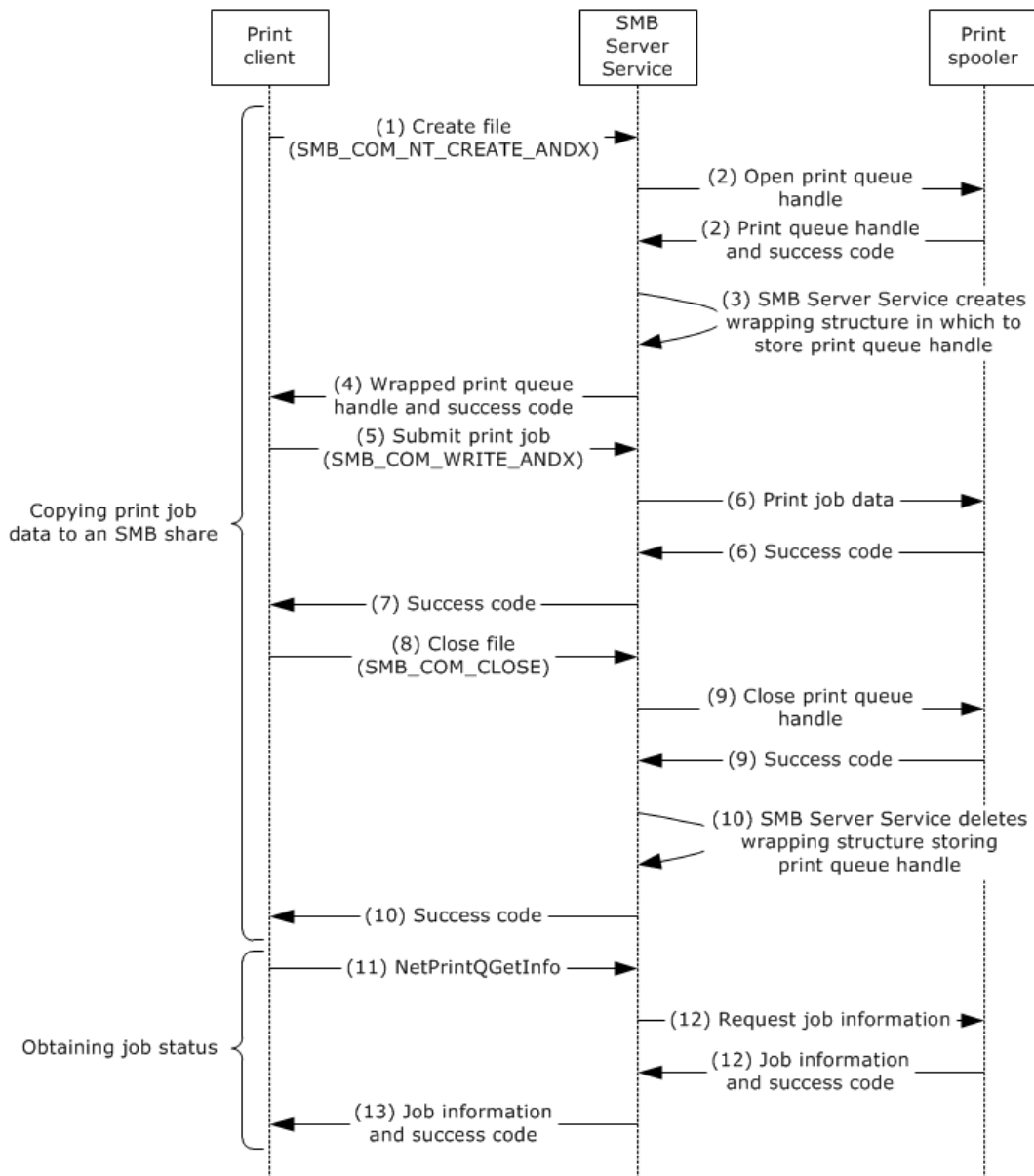
Figure 28: Print client sending a print job to an SMB share

Message groups	Description	References
Copying print job data to an SMB share	Using the SMB protocol, the print client copies print job data to an SMB share for redirection to the print queue.	[MS-SMB] (section 2.2)
Obtaining job status	Using the RAP protocol, the print client obtains the status of the print job.	[MS-RAP] (section 3.2.5.5)

The following steps describe this sequence.

1. The print client calls the Create File method (**SMB_COM_NT_CREATE_ANDX**) on the print server for the UNC name of the print queue.

2. The SMB Server Service on the print server redirects the call to a local API of the print spooler to open a handle to the print queue, and the local print spooler API returns the print queue handle and a success code to the SMB Server Service.

3. The SMB Server Service on the print server creates a wrapping structure in which to store the print queue handle and starts the print job by invoking a local print spooler API.

4. The SMB Server Service returns the handle of the wrapping structure and a success code to the print client as though it were a file handle.

5. The print client calls the method (**SMB_COM_WRITE_ANDX**) on the print server to submit the print job data to the file handle.

6. The SMB Server Service redirects the print job data to a local print spooler API using the print queue handle stored in the wrapping structure, and the local print spooler API returns a success code to the SMB Server Service.

7. The SMB Server Service returns a success code to the print client.

8. The print client ends the print job by calling the Close File method (**SMB_COM_CLOSE**) on the SMB Server Service.

9. The SMB Server Service calls a local print spooler API to close the print queue handle designated in the wrapping structure, and the local print spooler API returns a success code to the SMB Server Service.

10. The SMB Server Service frees the wrapping structure and then returns a success code to the print client.

    To then obtain information from the SMB Server Service about print job status, the user can enter the **net print** command at the command line,<26> to which the print client responds as follows:

11. The print client sends the **NetPrintQGetInfo** command defined by the RAP protocol [MS-RAP] to send the share name of the print queue on the print server.

12. The SMB Server Service on the print server redirects the call to a local API of the print spooler to obtain information about the jobs queued for the specified print queue, and the local print spooler API returns job information and a success code.

13. The SMB Server Service responds to the **NetPrintQGetInfo** request with print job information and a success code.

Final State

A new print job has been added to the print queue on the print server and is being sent to the associated port.

4 Microsoft Implementations

The information in this specification is applicable to the following versions of Windows:

- Windows NT Server 3.1 operating system

- Windows NT Server 3.5 operating system

- Windows NT Server 3.51 operating system

- Windows NT 4.0 operating system

- Microsoft Windows 98 operating system

- Windows 2000 operating system

- Windows 2000 Server operating system

- Windows Millennium Edition operating system

- Windows XP operating system

- Windows Server 2003 operating system

- Windows Server 2003 R2 operating system

- Windows Vista operating system

- Windows Server 2008 operating system

- Windows 7 operating system

- Windows Server 2008 R2 operating system

- Windows 8 operating system

- Windows Server 2012 operating system

- Windows 8.1 operating system

- Windows Server 2012 R2 operating system

Exceptions, if any, are noted in the following section. If a service pack number appears with the product version, behavior changed in that service pack. The new behavior also applies to subsequent service packs of the product unless otherwise specified.

4.1 Product Behavior

<1> Section 2.3.2: Windows implementations attempt to locate a driver from the following sources, in this order, when a driver cannot be obtained from the print server:

- The Print Store, which is the driver store in Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2 and the local driver cabinet file in earlier versions of Windows.

- Windows Update.

<2> Section 2.3.2: The maximum number of shared print queues allowed in a Windows workgroup is 256.

<3> Section 2.3.2: Group Policy settings can be applied to print clients and print servers running Windows 2000, Windows 2000 Server, Windows XP, Windows Server 2003, Windows Server 2003 R2, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2.

<4> Section 2.5.3.8.3: The net print command is no longer present in Windows 7, Windows 8, and Windows 8.1.

<5> Section 2.5.3.8.3: The net print command is no longer present in Windows 7, Windows 8, and Windows 8.1.

<6> Section 2.6: The Print Services System versions described in section 2.6 are supported in Windows releases as noted in the following table.

Print services system version	Print server release	Print client release	Method used by client to determine support level
PSS 1.0	Windows NT 3.1 Windows NT 3.5 Windows NT 3.51 Windows NT 4.0 Windows 2000 Windows 2000 Server Windows XP Windows Server 2003 Windows Server 2003 R2 Windows Server 2008 Windows Server 2008 R2 Windows Vista Windows 7 Windows 8 Windows Server 2012 Windows 8.1 Windows Server 2012 R2	Windows 98 Windows Millennium Edition Windows NT 3.1 Windows NT 3.5 Windows NT 3.51 Windows NT 4.0 Windows 2000 Windows 2000 Server Windows XP Windows Server 2003 Windows Server 2003 R2 Windows Server 2008 Windows Server 2008 R2 Windows Vista Windows 7 Windows 8 Windows Server 2012 Windows 8.1 Windows Server 2012 R2	The protocol described in [MS-RPRN] is not supported. The client detects this if the RPC binding to the endpoint for the protocol described in [MS-RPRN] fails.
PSS 2.0	Windows NT 3.1 Windows NT 3.5 Windows NT 3.51	Windows NT 3.1 Windows NT 3.5 Windows NT 3.51	The protocol described in [MS-RPRN] is supported except for the **RpcOpenPrinterEx** method.

Release: Thursday, February 13, 2014

Print services system version	Print server release	Print client release	Method used by client to determine support level
	Windows NT 4.0 Windows 2000 Windows 2000 Server Windows XP Windows Server 2003 Windows Server 2003 R2 Windows Server 2008 Windows Server 2008 R2 Windows Vista Windows 7 Windows 8 Windows Server 2012 Windows 8.1 Windows Server 2012 R2	Windows NT 4.0 Windows 2000 Windows 2000 Server Windows XP Windows Server 2003 Windows Server 2003 R2 Windows Server 2008 Windows Server 2008 R2 Windows Vista Windows 7 Windows 8 Windows Server 2012 Windows 8.1 Windows Server 2012 R2	
PSS 2.0 Additional capabilities described in [MS-RPRN]	Windows NT 4.0 Windows 2000 Windows 2000 Server Windows XP Windows Server 2003 Windows Server 2003 R2 Windows Server 2008 Windows Server 2008 R2 Windows Vista Windows 7 Windows 8 Windows Server 2012 Windows 8.1 Windows Server 2012 R2	Windows NT 4.0 Windows 2000 Windows 2000 Server Windows XP Windows Server 2003 Windows Server 2003 R2 Windows Server 2008 Windows Server 2008 R2 Windows Vista Windows 7 Windows 8 Windows Server 2012 Windows 8.1 Windows Server 2012 R2	The protocol described in [MS-RPRN] is supported including the **RpcOpenPrinterEx** method, and the OSVERSIONINFO.BuildNumber is less than or equal to 1381.
PSS 3.0	Windows 2000 Windows 2000 Server	Windows 2000 Windows 2000 Server	OSVERSIONINFO.BuildNumber is greater than 1381 but less than or equal to 3790.

Release: Thursday, February 13, 2014

Print services system version	Print server release	Print client release	Method used by client to determine support level
	Windows XP Windows Server 2003 Windows Server 2003 R2 Windows Server 2008 Windows Server 2008 R2 Windows Vista Windows 7 Windows 8 Windows Server 2012 Windows 8.1 Windows Server 2012 R2	Windows XP Windows Server 2003 Windows Server 2003 R2 Windows Server 2008 Windows Server 2008 R2 Windows Vista Windows 7 Windows 8 Windows Server 2012 Windows 8.1 Windows Server 2012 R2	
PSS 4.0	Windows Server 2008 Windows Server 2008 R2 Windows Server 2012 Windows Server 2012 R2	Windows Vista Windows 7 Windows 8 Windows 8.1	OSVERSIONINFO.BuildNumber is greater than 3790.

<7> Section 2.7.1.1: Only Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2 support this configuration.

<8> Section 2.9.1: This functionality requires Windows 2000, Windows XP, Windows Server 2003, Windows Server 2003 R2, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, or Windows Server 2012 R2.

<9> Section 3.1: Print client implementations on Windows 7, Windows 8, and Windows 8.1 can satisfy this precondition by disabling the PAR server endpoint on the print server, thus forcing the print server to use the Print System Remote Protocol for communication with the print client. The PAR server endpoint on the print server can be disabled through a registry setting on the print server.

<10> Section 3.1: In Windows, the package aware flag can be set on the printer driver using the registry key "HKLM\System\CurrentControlSet\Control\Print\Environments\<Environment name>\Drivers\Version-3\<print driver name>\PrintDriverAttributes". This key can take the following values:

Value	Description
0x00000001 PRINTER_DRIVER_PACKAGE_AWARE	The printer driver is part of a driver package.

Value	Description
0x00000005 PRINTER_DRIVER_SANDBOX_ENABLED	The printer driver supports print driver isolation.

Windows uses the Environment name strings "Windows NT x86", "Windows IA64", "Windows x64", and "Windows 4.0".

<11> Section 3.1: Print server support for the protocol described in [MS-PAR] can be disabled using a local Windows registry setting.

<12> Section 3.1: This precondition applies when the print client and print server are implemented on any Windows version except Windows 95, Windows 98, and Windows Millennium Edition. If the print server is implemented on Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2 a registry setting has to be used to disable the Print System Asynchronous Remote Protocol [MS-PAR].

<13> Section 3.1: The printer driver directory is not needed by the Windows print client to complete this scenario; the print client merely preemptively queries this piece of print server configuration data.

<14> Section 3.1: After registering for print status notifications, a Windows print client also calls the **RpcRouterRefreshPrinterChangeNotification** method on the print server with the *dwColor* parameter value 1. The print server then returns the **RpcRouterRefreshPrinterChangeNotification** method call with a success code. Until a Windows print client calls the **RpcRouterRefreshPrinterChangeNotification** method again with a different *dwColor* parameter value, a Windows print server will use 1 as the *dwColor* parameter value when calling the **RpcReplyOpenPrinterEx** method on the print client as described in task 3 of this example. As described in [MS-RPRN], Windows print clients and print servers use this mechanism to avoid order reversal of print status notifications due to network latency.

<15> Section 3.2: The maximum number of shared print queues allowed in a Windows workgroup is 256.

<16> Section 3.2: Print server support for the protocol described in [MS-PAR] can be disabled using a local Windows registry setting.

<17> Section 3.2: This precondition applies when the print client and print server are implemented on any Windows version except Windows 95, Windows 98, and Windows Millennium Edition. If the print server is implemented on Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, or Windows Server 2012 R2 a registry setting has to be used to disable the Print System Asynchronous Remote Protocol [MS-PAR].

<18> Section 3.2: The printer driver directory is not required by the Windows print client to complete this scenario; the print client merely preemptively queries this piece of print server configuration data.

<19> Section 3.2: After registering for print status notifications, a Windows print client also calls the **RpcRouterRefreshPrinterChangeNotification** method on the print server with the *dwColor* parameter value 1. The print server then returns the **RpcRouterRefreshPrinterChangeNotification** method call with a success code. Until a Windows print client calls the **RpcRouterRefreshPrinterChangeNotification** method again with a different *dwColor* parameter value, a Windows print server will use 1 as the *dwColor* parameter value when calling the **RpcReplyOpenPrinterEx** method on the print client as described in task 3 of this example. As described in [MS-RPRN], Windows print clients and print servers use this mechanism to avoid order reversal of print status notifications due to network latency.

<20> Section 3.3: The example described in this section applies when the print client is implemented on Windows Vista, Windows 7, Windows 8, or Windows 8.1, and the print server is implemented on Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, or Windows Server 2012 R2.

<21> Section 3.4: This precondition applies when the print client and print server are both implemented on any Windows version except Windows 95, Windows 98, and Windows Millennium Edition. If the print server is implemented on Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, or Windows Server 2012 R2 a registry setting is used to disable the Print System Asynchronous Remote Protocol [MS-PAR].

<22> Section 3.5: This precondition applies when the Administrative Client and print server are both implemented on any Windows version except Windows 95, Windows 98, and Windows Millennium Edition. If the print server is implemented on Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, or Windows Server 2012 R2 a registry setting is used to disable the Print System Asynchronous Remote Protocol [MS-PAR].

<23> Section 3.6: The example also applies to print clients running on Windows 95, Windows 98, and Windows Millennium Edition, as well as when a user sends a print job to a print queue by using the command line to copy a file to a printer share.

<24> Section 3.6: Windows 7, Windows 8, and Windows 8.1, print clients and Windows Server 2008 R2, Windows Server 2012, and Windows Server 2012 R2 print servers no longer support the copy /b or net print command line functionality.

<25> Section 3.6: The **net print** command has been deprecated in Windows 7, Windows 8, and Windows 8.1.

<26> Section 3.6: The **net print** command has been deprecated in Windows 7 Windows 8, and Windows 8.1.

5   Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

6   Index

A

C

D

E

F

G

H

I

Release: Thursday, February 13, 2014

T

U

V