

[MS-TSWP]: Terminal Services Workspace Provisioning Protocol Specification

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's Open Specification Promise (available here: <http://www.microsoft.com/interop/osp>) or the Community Promise (available here: <http://www.microsoft.com/interop/cp/default.mspx>). 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 iplq@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.
- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail 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.

Revision Summary

Date	Revision History	Revision Class	Comments
12/05/2008	0.1	Major	Initial Availability
01/16/2009	0.2	Minor	Updated the technical content.
02/27/2009	0.2.1	Editorial	Revised and edited the technical content.
04/10/2009	1.0	Major	Updated and revised the technical content.
05/22/2009	1.0.1	Editorial	Revised and edited the technical content.
07/02/2009	2.0	Major	Updated and revised the technical content.
08/14/2009	2.0.1	Editorial	Revised and edited the technical content.
09/25/2009	2.1	Minor	Updated the technical content.
11/06/2009	2.2	Minor	Updated the technical content.
12/18/2009	3.0	Major	Updated and revised the technical content.
01/29/2010	3.1	Minor	Updated the technical content.
03/12/2010	4.0	Major	Updated and revised the technical content.
04/23/2010	5.0	Major	Updated and revised the technical content.
06/04/2010	6.0	Major	Updated and revised the technical content.
07/16/2010	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
08/27/2010	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
10/08/2010	6.0	No change	No changes to the meaning, language, or formatting of the technical content.
11/19/2010	7.0	Major	Significantly changed the technical content.
01/07/2011	7.0	No change	No changes to the meaning, language, or formatting of the technical content.
02/11/2011	7.0	No change	No changes to the meaning, language, or formatting of the technical content.

Contents

1 Introduction	5
1.1 Glossary	5
1.2 References	6
1.2.1 Normative References	6
1.2.2 Informative References	6
1.3 Overview	7
1.4 Relationship to Other Protocols	7
1.5 Prerequisites/Preconditions	7
1.6 Applicability Statement	7
1.7 Versioning and Capability Negotiation	7
1.8 Vendor-Extensible Fields	8
1.9 Standards Assignments	8
2 Messages	9
2.1 Transport	9
2.2 Message Syntax	9
2.2.1 Resource List Syntax	9
2.2.1.1 Schema Version 1.1	9
2.2.2 Schema Element Definitions	12
2.2.2.1 Schema Version 1.1 Element Definitions	12
2.2.2.1.1 ResourceCollection Element	12
2.2.2.1.2 Publisher Element	12
2.2.2.1.3 Resources Element	13
2.2.2.1.4 Resource Element	13
2.2.2.1.5 Icons Element	13
2.2.2.1.6 Icon Elements	13
2.2.2.1.7 HostingTerminalServers Element	14
2.2.2.1.8 HostingTerminalServer Element	14
2.2.2.1.9 ResourceFile Element	14
2.2.2.1.10 TerminalServerRef Element	14
2.2.2.1.11 TerminalServers Element	14
2.2.2.1.12 TerminalServer Element	14
2.2.2.1.13 FileExtensions Element	15
2.2.2.1.14 FileExtension Element	15
2.2.3 .ASPXAUTH Cookie	15
2.2.4 Resources	15
3 Protocol Details	17
3.1 Common Details	17
3.1.1 Abstract Data Model	19
3.1.1.1 Authentication Cookie	19
3.1.1.2 XML Files	19
3.1.1.3 Icon Files	20
3.1.1.4 Resource Files	20
3.1.1.5 Resources	20
3.2 Client Details	20
3.2.1 Abstract Data Model	20
3.2.2 Timers	20
3.2.3 Initialization	20
3.2.4 Higher-Layer Triggered Events	20

3.2.5	Message Processing Events and Sequencing Rules	20
3.2.6	Timer Events	21
3.2.7	Other Local Events	21
3.3	Server Details	21
3.3.1	Abstract Data Model	21
3.3.1.1	Authentication Cookie	21
3.3.2	Timers	21
3.3.3	Initialization	21
3.3.4	Higher-Layer Triggered Events	21
3.3.5	Message Processing Events and Sequencing Rules	21
3.3.6	Timer Events	21
3.3.7	Other Local Events	21
4	Protocol Examples	22
4.1	Schema Version 1.1 Examples	22
4.1.1	Message with One Hosting Terminal Server	22
4.1.2	Message with Multiple Terminal Servers	23
4.2	.ASPXAUTH Cookie Message Returned from the Server	26
5	Security	28
5.1	Security Considerations for Implementers	28
5.2	Index of Security Parameters	28
6	Appendix A: Product Behavior	29
7	Change Tracking	30
8	Index	31

1 Introduction

This is a specification of the Terminal Services Workspace Provisioning Protocol.

The Terminal Services Workspace Provisioning Protocol is used to discover and provision workspaces by transferring remote resource information from a server to a client. The client can use this resource information to launch resources such as **remote applications** on a remote server.

1.1 Glossary

The following terms are defined in [\[MS-GLOS\]](#):

- authentication (2)**
- binary large object (BLOB)**
- domain name (1)**
- globally unique identifier (GUID)**
- Hypertext Transfer Protocol (HTTP)**
- Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)**
- Kerberos**
- NT LAN Manager (NTLM) Authentication Protocol**
- remote application**
- Remote Desktop Protocol (RDP)**
- schema**
- Secure Sockets Layer (SSL)**
- Simple and Protected GSS-API Negotiation Mechanism (SPNEGO)**
- terminal server**
- Terminal Services**
- Transmission Control Protocol (TCP)**
- Transport Layer Security (TLS)**
- Uniform Resource Locator (URL)**
- XML**
- XML namespace**
- XML schema (XSD)**

The following terms are specific to this document:

client: The entity that initiates the **HTTP** connection.

Content-Type: A property of an **HTTP** message, specified in the message header, which defines the type of data in the message payload. The Content Type header is defined in [\[RFC2616\]](#) section 14.17.

HTTPS proxy: An intermediary program that acts as both a **server** and a **client** for the purpose of making requests on behalf of other **clients**, tunneled using **Secure Sockets Layer (SSL)** or **Transport Layer Security (TLS)** for providing secure, encrypted communication.

publisher: A set of resources that are contained in the same **workspace**.

server: The entity that responds to the **HTTP** connection.

TSWPP: The Terminal Services Workspace Provisioning Protocol.

workspace: A set of remote resources, such as **remote applications** and desktops, which are published to end users.

XML resource list: An **XML** file that is sent from a **TSWPP server** to a **TSWPP client**. This file, which conforms to the **XML schema (XSD)** in section [2.2.1](#), describes a set of remote resources.

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as described in [\[RFC2119\]](#). All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 References

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information. Please check the archive site, <http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624>, as an additional source.

[RFC2109] Kristol, D., and Montulli, L., "HTTP State Management Mechanism", RFC 2109, February 1997, <http://www.ietf.org/rfc/rfc2109.txt>

[RFC2616] Fielding, R., Gettys, J., Mogul, J., et al., "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999, <http://www.ietf.org/rfc/rfc2616.txt>

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

[RFC4559] Jaganathan, K., Zhu, L., and Brezak, J., "SPNEGO-based Kerberos and NTLM HTTP Authentication in Microsoft Windows", RFC 4559, June 2006, <http://www.ietf.org/rfc/rfc4559.txt>

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.ietf.org/rfc/rfc2119.txt>

1.2.2 Informative References

[MS-GLOS] Microsoft Corporation, "[Windows Protocols Master Glossary](#)", March 2007.

[MS-RDPBCGR] Microsoft Corporation, "[Remote Desktop Protocol: Basic Connectivity and Graphics Remoting Specification](#)", June 2007.

[MSDN-TSCCRDP] Microsoft Corporation, "Terminal Services Client Configuration through the .rdp File", <http://msdn.microsoft.com/en-us/library/aa915001.aspx>

[NTLM] Microsoft Corporation, "Microsoft NTLM", <http://msdn.microsoft.com/en-us/library/aa378749.aspx>

If you have any trouble finding [NTLM], please check [here](#).

[XML10/4] W3C Recommendation, "Extensible Markup Language (XML) 1.0 (Fourth Edition)", August 16, 2006, <http://www.w3.org/TR/2006/REC-xml-20060816>

[XMLSCHEMA1.1/1] Thompson, H.S., Ed., Sperberg-McQueen, C.M., Ed., Mendelsohn, N., Ed., Beech, D., Ed., and Maloney, M., Ed., "XML Schema 1.1 Part 1: Structures", W3C Working Draft, March 2006, <http://www.w3.org/TR/2006/WD-xmlschema11-1-20060330/>

[XMLSCHEMA1.1/2] Peterson, D., Ed., Biron, P.V., Ed., Malhotra, A., Ed., and Sperberg-McQueen, C.M., Ed., "XML Schema 1.1 Part 2: Datatypes", W3C Working Draft, June 2008, <http://www.w3.org/TR/2008/WD-xmlschema11-2-20080620/>

1.3 Overview

The Terminal Services Workspace Provisioning Protocol (**TSWPP**) was created to provide users with a unified view of the resources that have been made available to them by an administrator. This is needed when a user wants to use their resources from an unmanaged computer, because administrators typically do not have access to unmanaged computers. This protocol obtains information that a client machine can use to launch a remote application that runs on a **server** or a virtual machine (VM). The client can arrange the remote resources so that a user can use a pointing device to launch the application.

TSWPP specifies an **HTTP**-based interface [\[RFC2616\]](#) that uses **XML** messages to describe the remote resources that are available. In this specification, the entity that initiates the HTTP connection is referred to as the **client**, and the entity that responds to the HTTP connection is referred to as the server.

1.4 Relationship to Other Protocols

TSWPP depends on HTTP [\[RFC2616\]](#) [\[RFC2109\]](#) and **HTTPS** [\[RFC2818\]](#) to transfer all protocol messages, including resource information. Any version of HTTP can be used with TSWPP.

For initial **authentication**, TSWPP depends on the authentication scheme defined in [\[RFC4559\]](#).

1.5 Prerequisites/Preconditions

The following are prerequisites for using TSWPP:

- TSWPP does not provide a mechanism for a client to discover the **Uniform Resource Locator (URL)** to the server; thus, the client is required to have a valid URL to the server. This URL provides a path either to the **XML resource list** or to the entry point that the server uses to generate the XML resource list for the client.
- Both client and server implementations of TSWPP are present and running.
- The client machine has the necessary applications to launch any of the resource files contained in the XML resource file. For example, if one of the resource files in the XML resource file is a **Remote Desktop Protocol (RDP)** configuration file [\[MSDN-TSCCRDP\]](#), then the **Terminal Services** client is required to be present in order to launch the application, and that client will use the RDP protocol [\[MS-RDPBCGR\]](#) to connect.

1.6 Applicability Statement

TSWPP is applicable when a client requires the constituent elements of an application that is located on a remote server in order to execute that application remotely on the server machine. These elements can include, but are not limited to, icons, remote files, and information describing the remotely-executing resource or application.

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

- **Supported Transports:** TSWPP is implemented on HTTP (section [2.1](#)).
- **Protocol Versions:** Servers specify the protocol version by using the **SchemaVersion** attribute in the XML file. Clients **MUST** support the version specified by the server because TSWPP does not support protocol negotiation.

- **Security and Authentication Methods:** TSWPP supports HTTP access authentication, as specified in [\[RFC4559\]](#).
- **Localization:** This specification does not define any locale-specific protocol behavior.
- **Capability Negotiation:** TSWPP does not support version negotiation.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

Parameter	Value	Reference
TCP port	443	[RFC2818]
XML namespace	"http://schemas.microsoft.com/ts/2007/05/tswf"	[XML10/4]

2 Messages

2.1 Transport

TSWPP uses HTTP protocol messages that are carried in the HTTP message headers and message body, as specified in [\[RFC2616\]](#) and [\[RFC2109\]](#). TSWPP uses HTTPS to transport these messages, as specified in [\[RFC2818\]](#).

A TCP port has not been reserved for TSWPP. Administrators often use TCP port 443 when setting up their Web servers, because many **HTTPS proxy** servers forward only HTTPS traffic that uses port 443. TSWPP can use any port as defined in [\[RFC2818\]](#) section 2.3 "Port Number".

TSWPP uses the access authentication functionality of the HTTP layer. The supported HTTP access authentication schemes are implementation-specific, as specified in [\[RFC4559\]](#).

2.2 Message Syntax

2.2.1 Resource List Syntax

This section specifies the complete XML **schema**, the set of attributes and object classes that govern the creation and update of objects, (see [\[XMLSCHEMA1.1/1\]](#) and [\[XMLSCHEMA1.1/2\]](#) for details on XML schemas) for messages that are sent from the TSWPP server to the TSWPP client, which defines the syntax for XML resource lists.

The **targetNamespace** listed here (<http://schemas.microsoft.com/ts/2007/05/tswf>) is a URI (see <http://www.w3.org/tr/REC-xml-names/> and <http://www.ietf.org/rfc/rfc3986.txt> for details on XML namespace URIs), and is just meant to be used as a string identifier for the TSWPP namespace.

2.2.1.1 Schema Version 1.1

The schema version for this XML schema is 1.1 and is listed in the **version** attribute of the `<xs:schema>` element example that follows.

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema targetNamespace="http://schemas.microsoft.com/ts/2007/05/tswf"
  elementFormDefault="qualified" xmlns="http://schemas.microsoft.com/ts/2007/05/tswf"
  xmlns:mstns="http://schemas.microsoft.com/ts/2007/05/tswf"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" version="1.1">
  <xs:element name="ResourceCollection" type="ResourceCollectionType" />
  <xs:complexType name="ResourceCollectionType">
    <xs:sequence>
      <xs:element name="Publisher" type="PublisherType" minOccurs="1" maxOccurs="unbounded">
        <xs:key name="ResourceIDKey">
          <xs:selector xpath="mstns:Resources/mstns:Resource" />
          <xs:field xpath="@ID" />
        </xs:key>
        <xs:key name="TerminalServerIDKey">
          <xs:selector xpath="mstns:TerminalServers/mstns:TerminalServer" />
          <xs:field xpath="@ID" />
        </xs:key>
        <xs:keyref name="ResourceToTerminalServerRef" refer="TerminalServerIDKey">
          <xs:selector
            xpath="mstns:Resources/mstns:Resource/mstns:HostingTerminalServers/mstns:HostingTerminalServer/mstns:TerminalServerRef" />
          <xs:field xpath="@Ref" />
        </xs:keyref>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

```

        </xs:keyref>
    </xs:element>
</xs:sequence>
<xs:attribute name="SchemaVersion" type="xs:string" use="required" />
<xs:attribute name="PubDate" type="xs:dateTime" />
<xs:anyAttribute />
</xs:complexType>
<xs:complexType name="PublisherType">
    <xs:sequence>
        <xs:element name="Resources">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="Resource" type="ResourceType" minOccurs="0"
maxOccurs="unbounded" />
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="TerminalServers">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="TerminalServer" type="TerminalServerType" minOccurs="0"
maxOccurs="unbounded" />
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:any minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="LastUpdated" type="xs:dateTime" />
    <xs:attribute name="Name" type="xs:string" />
    <xs:attribute name="ID" type="xs:string" />
    <xs:attribute name="Description" type="xs:string" />
    <xs:anyAttribute />
</xs:complexType>
<!-- Resource and related Types -->
<xs:complexType name="ResourceType">
    <xs:sequence>
        <xs:element name="Icons" minOccurs="0" maxOccurs="1">
            <xs:complexType>
                <xs:all>
                    <xs:element name="Icon16" type="IconType" minOccurs="0" maxOccurs="1" />
                    <xs:element name="Icon32" type="IconType" minOccurs="0" maxOccurs="1" />
                    <xs:element name="Icon48" type="IconType" minOccurs="0" maxOccurs="1" />
                    <xs:element name="Icon64" type="IconType" minOccurs="0" maxOccurs="1" />
                    <xs:element name="Icon100" type="IconType" minOccurs="0" maxOccurs="1" />
                    <xs:element name="Icon256" type="IconType" minOccurs="0" maxOccurs="1" />
                    <xs:element name="IconRaw" type="IconType">
                        </xs:element>
                    </xs:all>
                </xs:complexType>
            </xs:element>
            <xs:element name="FileExtensions" minOccurs="0" maxOccurs="1">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="FileExtension" type="FileExtensionType" minOccurs="0"
maxOccurs="unbounded" />
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <xs:element name="HostingTerminalServers">

```

```

        <xs:complexType>
          <xs:sequence>
            <xs:element name="HostingTerminalServer" type="HostingTerminalServerType"
minOccurs="0" maxOccurs="unbounded" />
          </xs:sequence>
          <xs:anyAttribute />
        </xs:complexType>
      </xs:element>
      <xs:any minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="ID" type="xs:string" use="required" />
    <xs:attribute name="Alias" type="xs:string" />
    <xs:attribute name="Title" type="xs:string" />
    <xs:attribute name="LastUpdated" type="xs:dateTime" />
    <xs:attribute name="Type">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="Desktop" />
          <xs:enumeration value="RemoteApp" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="RequiredCommandLine" type="xs:string" />
    <xs:attribute name="ExecutableName" type="xs:string" />
    <xs:anyAttribute />
  </xs:complexType>
  <xs:complexType name="IconType">
    <xs:sequence>
      <xs:element name="FileContent" type="xs:string" minOccurs="0" maxOccurs="1" />
    </xs:sequence>
    <xs:attribute name="Dimensions" type="xs:string" />
    <xs:attribute name="FileType" type="xs:string" />
    <xs:attribute name="FileURL" type="xs:string" />
    <xs:attribute name="Index" type="xs:integer" />
    <xs:anyAttribute />
  </xs:complexType>
  <xs:complexType name="FileExtensionType">
    <xs:sequence>
      <xs:any minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="Name" type="xs:string" />
    <xs:anyAttribute />
  </xs:complexType>
  <xs:complexType name="TerminalServerRefType">
    <xs:attribute name="Ref" type="xs:string" use="required" />
  </xs:complexType>
  <xs:complexType name="ResourceFileType">
    <xs:sequence>
      <xs:element name="Content" type="xs:string" minOccurs="0" maxOccurs="1" />
    </xs:sequence>
    <xs:attribute name="URL" type="xs:string" />
    <xs:attribute name="FileExtension" type="xs:string" default=".rdp" />
    <xs:anyAttribute />
  </xs:complexType>
  <xs:complexType name="HostingTerminalServerType">
    <xs:sequence>
      <xs:element name="ResourceFile" type="ResourceFileType" minOccurs="0" maxOccurs="1" />
      <xs:element name="TerminalServerRef" type="TerminalServerRefType" />
      <xs:any minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>

```

```

    </xs:sequence>
    <xs:anyAttribute />
</xs:complexType>
<!-- TerminalServer and related Types -->
<xs:complexType name="TerminalServerType">
  <xs:attribute name="ID" type="xs:string" use="required" />
  <xs:attribute name="Name" type="xs:string" />
  <xs:attribute name="LastUpdated" type="xs:dateTime" />
  <xs:anyAttribute />
</xs:complexType>
</xs:schema>

```

2.2.2 Schema Element Definitions

2.2.2.1 Schema Version 1.1 Element Definitions

This section specifies the elements in the **XML schema (XSD)** that are defined in section [2.2.1.1](#). The values of all attributes MUST be **xs:string**, unless specified otherwise.

2.2.2.1.1 ResourceCollection Element

The <ResourceCollection> element contains all the other elements defined in section [2.2.2.1](#). The server MUST only put one **publisher** element as defined in section [2.2.2.1.2](#). The <ResourceCollection> element defines the following attributes:

- **PubDate:** The publication date. This date SHOULD be included in the <ResourceCollection> element and MUST be in the form **xs:dateTime**.
- **SchemaVersion:** The version number of the XML schema. This attribute MUST be included in the <ResourceCollection> element and SHOULD be based on the version of the schema described in section [2.2.1.1](#) of this document. The schema version for the XML schema listed in section [2.2.1.1](#) of this document is schema version 1.1. The schema version number can be found in the **version** attribute of the <xs:schema> element in the XML schema (section [2.2.1](#)).

2.2.2.1.2 Publisher Element

The <Publisher> element contains all the attributes for one publisher. It defines the following attributes:

- **LastUpdated:** A timestamp that indicates the date and time of any changes made by any entries under the <Publisher> element. This timestamp MUST be in **xs:dateTime** format.
- **Name:** The name of the publisher of the resources.
- **ID:** A **GUID**, as defined in [\[MS-DTYP\]](#) sections [2.3.2](#), [2.3.2.2](#), and [2.3.2.3](#), or the fully qualified **domain name**, a name with a structure indicated by dots, of the server that provides the XML file. The value of this attribute MUST be globally unique.
- **Description:** An optional description of the publisher.
- **TerminalServerIDKey:** This declares that the **ID** attribute of the <TerminalServer> element (section [2.2.2.1.12](#)) MUST be unique among the <TerminalServer> elements contained in the <TerminalServer> element (section [2.2.2.1.11](#)). The ID attribute is defined in section [2.2.2.1.12](#).

- **ResourceToTerminalServerRef:** This declares that references to a <TerminalServer> element by ID SHOULD be contained in <TerminalServerRef> elements. The link to the <TerminalServer> ID is in the **Ref** attribute as defined in section [2.2.2.1.10](#).

2.2.2.1.3 Resources Element

The <Resources> element contains all of the resources that have been published by a publisher as specified in section [2.2.2.1.2](#).

2.2.2.1.4 Resource Element

The <Resource> element describes one resource such as an application that can be launched remotely. It contains all information that a client requires to display and launch the application. This element defines the following attributes:

- **ID:** A unique identifier for the resource. This identifier MUST be unique among all other resource identifiers within the containing <Resources> element (section [2.2.2.1.3](#)). The unique identifier SHOULD be generated in such a manner that the identifier remains the same on each request for the XML file.
- **Alias:** An identifier that the server uses to look up the requested resource. This identifier MUST be unique among all other resource identifiers within the containing <Resources> element (section [2.2.2.1.3](#)). The unique identifier SHOULD be generated in such a manner that the identifier remains the same on each request for the XML file.
- **Title:** The name of the resource. The client SHOULD use the value of this attribute to display the resource name to the user.
- **LastUpdated:** An optional timestamp, which specifies the last time that anything changed in this resource. The timestamp MUST be in **xs:dateTime** format.
- **Type:** The type of resource that this <Resource> element specifies. This type MUST be either "Desktop" or "RemoteApp". If the resource will start a complete desktop on a remote computer, use the "Desktop" type; otherwise, use the "RemoteApp" type.
- **ExecutableName:** This attribute specifies the executable that runs on the server if this resource is launched.
- **RequiredCommandLine:** This attribute is ignored by the client and MUST NOT be used by the server.

2.2.2.1.5 Icons Element

The <Icons> element contains all icons associated with one resource.

2.2.2.1.6 Icon Elements

The Icon elements specify icons for a resource or application. Icon elements include <Icon16>, <Icon32>, <Icon48>, <Icon64>, <Icon100>, <Icon256>, and <IconRaw>. The numeric part of the name specifies that the image used for the icon MUST be square, with height and width equal to that number. For example, an icon with the name **Icon256** must have a height and width of 256 pixels. The <IconRaw> element can contain multiple icons with different sizes.

Icon elements define the following attributes:

- **Dimensions:** An attribute that specifies the height and width of the icon with an "x" separating the two entries, such as "32x32". This attribute SHOULD appear only when the icon name is a value that contains a specific size such as **Icon16** and SHOULD NOT appear when the icon name is **IconRaw**.
- **FileType:** The file extension for the file type of the image. For example, if the image file on the server is named Paint.ico, then the **FileType** attribute would be "Ico".
- **FileURL:** An optional URL that specifies the location of the image file. This attribute MUST be present if the icon is available from a Web server.
- **FileContent:** This element is ignored by the client and MUST NOT be used by the server.
- **Index:** This attribute is ignored by the client and MUST NOT be used by the server.

2.2.2.1.7 HostingTerminalServers Element

The <HostingTerminalServers> element contains elements for all the **terminal servers**, computers on which Terminal Services is running, that are hosting a resource. This element MUST contain at least one [HostingTerminalServer](#) element, as defined in section [2.2.2.1.8](#).

2.2.2.1.8 HostingTerminalServer Element

The <HostingTerminalServer> element contains the name of the file required to launch a remote resource as defined in section [2.2.2.1.9](#) and an element that links to the terminal server reference, as defined in section [2.2.2.1.10](#).

2.2.2.1.9 ResourceFile Element

The <ResourceFile> element contains the name of the file that is required in order to launch a remote resource on a server. This element defines the following attributes:

- **FileExtension:** The file extension of the resource file that is defined in this resource. This file extension is what the client SHOULD use to determine how to launch the resource file. For example, if the file is for the terminal server client, then this file extension MUST be ".rdp".
- **URL:** The URL to the location of this resource file. This attribute MUST be present if the resource file is available from a Web server.
- **Content:** This element is ignored by the client and MUST NOT be used by the server.

2.2.2.1.10 TerminalServerRef Element

The <TerminalServerRef> element is a link element that contains one attribute: **Ref**. This attribute is the unique identifier for the terminal server as defined in section [2.2.2.1.12](#).

2.2.2.1.11 TerminalServers Element

The <TerminalServers> element contains <TerminalServer> elements (section [2.2.2.1.12](#)) for all terminal servers that host the remote resources defined in section [2.2.2.1.3](#).

2.2.2.1.12 TerminalServer Element

The <TerminalServer> element contains the definition for one hosting terminal server. It defines the following attributes:

- **ID:** A reference as defined in section [2.2.2.1.10](#). This attribute value MUST be unique among all terminal server elements contained in the <TerminalServers> element (section [2.2.2.1.11](#)).
- **Name:** The name of the terminal server, which SHOULD be the fully qualified domain name of that server.
- **LastUpdated:** The timestamp, which specifies the last time anything changed in this <TerminalServer> element. This entry MUST be in **xs:dateTime** format.

2.2.2.1.13 FileExtensions Element

The <FileExtensions> element is not implemented and MUST be set as empty by the server, for example, <FileExtensions>.

2.2.2.1.14 FileExtension Element

The <FileExtension> element is not implemented and MUST not be put in the TSWPP XML feed.

2.2.3 .ASPXAUTH Cookie

If a user's interactions with the HTML login URL have allowed the TSWPP server to establish the user's identity, the remote server SHOULD generate a cookie that identifies the user and allows authentication to the server. The contents of the cookie SHOULD be signed and encrypted. The specific implementation of this cookie including the signing and encryption algorithms is dependent on the implementation of the TSWPP server, because only the server is required to parse the contents of the cookie. If the server implements the cookie, then the cookie MUST be returned in an HTTP payload with a **Content-Type** of "application/x-msts-webfeed-login".

2.2.4 Resources

Resources are stored on the TSWPP server in memory and comprise the list of resources that are used to generate the [XML files \(section 3.1.1.2\)](#) that are sent to the TSWPP client. The data stored for a resource includes any [icon files \(section 3.1.1.3\)](#) associated with that resource as well as the [resource file \(section 3.1.1.4\)](#) itself. This data includes a friendly display name for the resource, as well as a unique identifier.

A client MUST be able to authenticate to the TSWPP server before the client can retrieve the XML file (section 3.1.1.2) containing the list of resources. A client MUST request the XML file from the TSWPP server, and the server MUST respond to requests from a client for the XML file containing the list of resources, as specified in section [2](#). The mechanism by which the XML file (section 3.1.1.2) contents are assembled is server implementation-specific. The server SHOULD receive a user identity as shown in the figures in section [3.1](#), and it SHOULD use that identity to create a customized XML resource list for that user. How the XML resource list is customized or if is customized at all is up to the server implementation. This is done by including only <ResourceFile> elements that the user can access.

That is because each user can have a different XML resource list returned from a TSWPP server if the server is implemented to customize the XML resource list for each user. The server SHOULD cache the information used to generate the XML file (section 3.1.1.2), so that the server can quickly respond to requests for the list of resources.

If the XML resource list that the client requests from the server is not available, the server MUST return an HTTP "404 Not Found" error code [\[RFC2616\]](#).

The client implementation makes the resources available to the user by installing them in a manner that allows easy user access to the resources. In addition, the client implementation stores the XML resource list locally and update it at least every 24 hours through new TSWPP transactions.

The client SHOULD make efforts to keep the locally stored XML resource list up-to-date. The client SHOULD accomplish this by periodically retrieving a new copy of the resource list from the server. The method of triggering this periodic retrieval of the XML resource list is specific to the client implementation. The client MAY trigger retrieval of the XML resource list by using a periodic task scheduler provided by the client operating system.

The client SHOULD prevent resource files with unsafe file extensions from being downloaded. This SHOULD be done before downloading by checking the FileExtension attribute of the <ResourceFile> element section [2.2.2.1.9](#) and determining whether the file extension listed in that attribute is registered as being safe. Which file extensions are considered safe and the manner in which they are registered as safe for download with the client is a detail of the client implementation. The server has no concept of safe or unsafe file extensions.

If the server returns an HTTP "404 Not Found" error code [\[RFC2616\]](#) when the client is connecting on the initial attempt to set up a **workspace**, the client SHOULD NOT set up a workspace and SHOULD warn the user that the connection could not be made. If the server returns an HTTP "404 Not Found" error when the client is connecting in order to update workspace resources that were previously downloaded, the client SHOULD log a warning but leave the existing resources untouched.

3 Protocol Details

3.1 Common Details

The message flow between a TSWPP client and TSWPP server on the first connection is shown in the following figure.

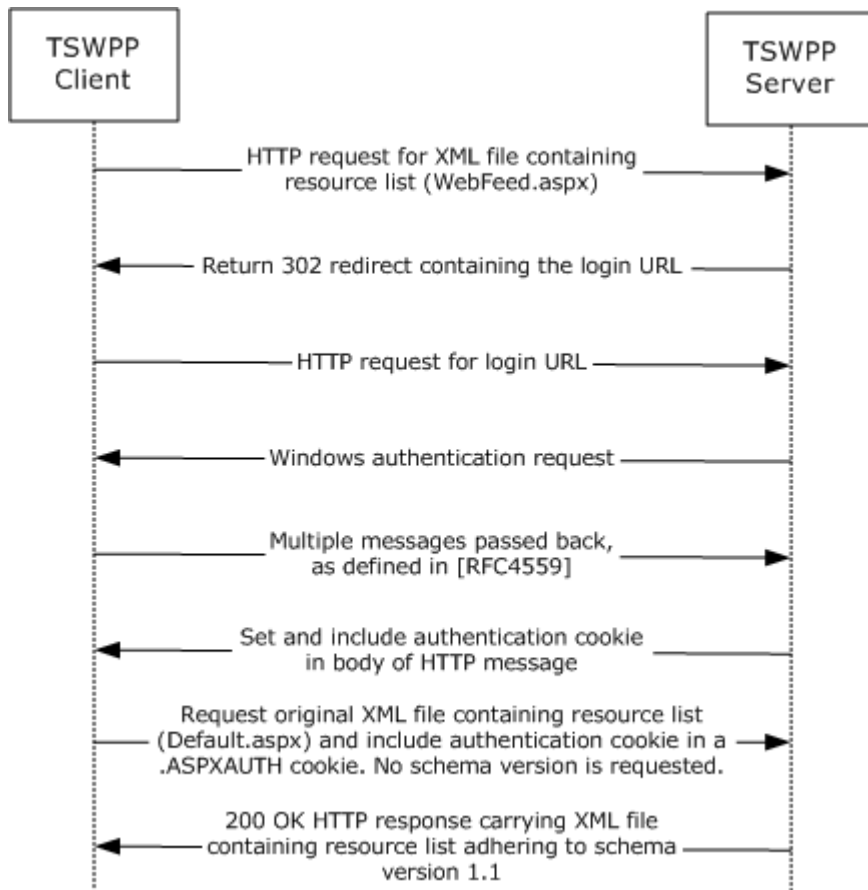


Figure 1: Message flow between a TSWPP client and a TSWPP server on the first connection

In the above figure, the TSWPP client MUST send an HTTP request for the XML file containing the resource list. The XML resource list MUST conform to the XML schema (XSD) for messages that are sent from a TSWPP server to a TSWPP client. Section [2.2.1](#) specifies the syntax for XML resource lists. Because the TSWPP client did not include the .ASPXAUTH cookie in its request, the client is not authenticated and the server SHOULD send a 302 redirect code with the login URL for the client to authenticate. If the client receives the 302 packet from the server, then the client MUST send an HTTP request to the login URL provided by the server. The server SHOULD then redirect the client to the login URL. This login URL SHOULD initiate the authentication process, and that SHOULD be achieved by using **Simple and Protected GSS-API Negotiation Mechanism (SPNEGO)**-based **Kerberos** and **NT LAN Manager (NTLM) Authentication Protocol** over HTTP, as specified in [\[RFC4559\]](#). For more information about the authentication process, see [\[NTLM\]](#). The specific authentication mechanism is negotiated between the server and the client, and thus results in multiple messages being passed between the client and server to complete authentication as

defined in [\[RFC4559\]](#). The authentication cookie SHOULD be obtained in the initial request to the server and MUST be carried in the HTTP message body, as specified in [\[RFC2616\]](#) and [\[RFC2109\]](#). If the server provides an authentication cookie in the previous step, the authentication cookie is treated as an opaque blob and MUST be sent by the client to the server in all subsequent requests and MUST be carried in the HTTP message headers, as specified in [\[RFC2616\]](#) and [\[RFC2109\]](#).

The message flow between a client and server on subsequent requests for the XML file containing the resource list is shown in the following figure.

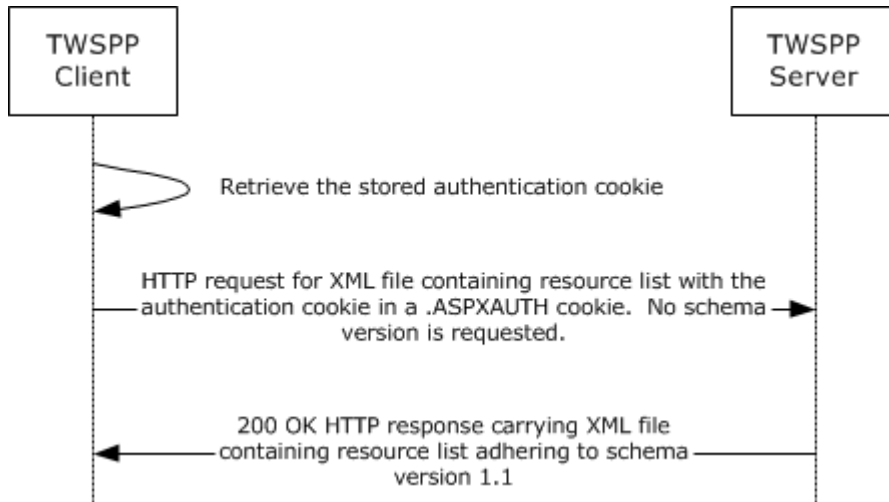


Figure 2: Message flow between a TSWPP client and a TSWPP server on subsequent requests for the XML file containing the resource list

In the above figure, the TSWPP client has already obtained the .ASPXAUTH cookie as shown in the first figure depicting message flow between a TSWPP client and TSWPP server. This cookie is set in the HTTP request header (section [2.2.3](#)) and the XML file containing the resource list is requested. The XML resource list MUST conform to the XML schema (XSD) for messages that are sent from a TSWPP server to a TSWPP client. Section [2.2.1](#) specifies the syntax for XML resource lists. The server SHOULD authenticate the user's access by using the .ASPXAUTH cookie (section [3.3.1.1](#)) if the server provides a cookie authentication mechanism and MUST return the XML file to the client.

The message flow between a client and server for requests for icon and resource file URLs that were specified in the XML file containing the resource list is shown in the following diagram. Icon and resource file URLs are specified in the XML schema definitions for Icon Elements (section [2.2.2.1.6](#)) and the ResourceFile Element (section [2.2.2.1.9](#)).

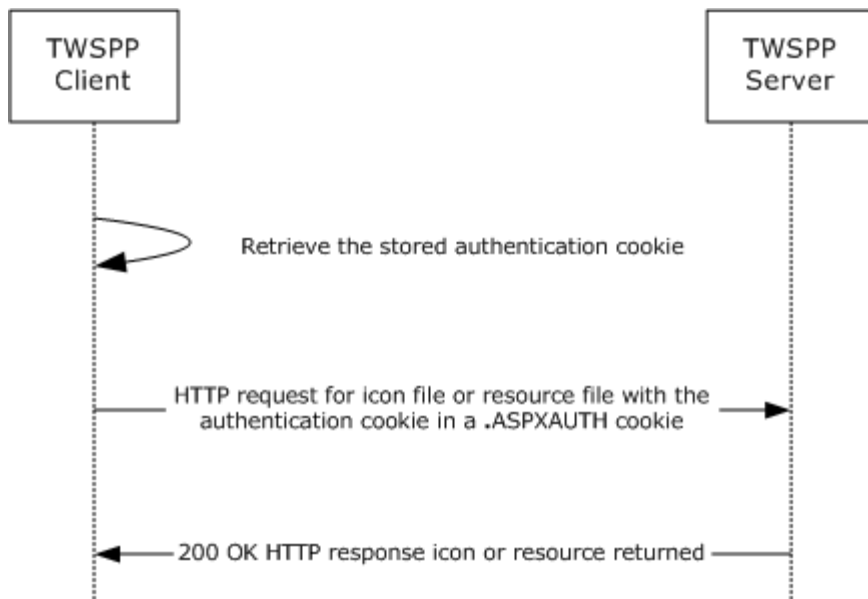


Figure 3: Message flow between a TSWPP client and a TSWPP server on requests for resource and icon file URLs

In the above figure, the XML file containing the list of resources has already been downloaded. The client is downloading the resource files and icon files that are defined in the XML file. The client MUST set the .ASPXAUTH cookie if the server provided an authentication cookie and the client MUST request the resource and icon files by using the resources that were returned in the XML file. If the requested resource or icon does not exist then the server MUST return an HTTP "404 Not Found" error code [\[RFC2616\]](#).

3.1.1 Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. The described organization is provided to facilitate the explanation of how the protocol behaves. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that described in this document.

3.1.1.1 Authentication Cookie

When a TSWPP client first connects to the TSWPP server, an authentication cookie is transmitted to the client, as specified in section [3.1](#). This opaque server generated cookie contains information to authenticate a user to the server as well as the security identifier (SID), as defined in [\[MS-DTYP\]](#) section 2.4.2, for the user that originally authenticated to the server.

3.1.1.2 XML Files

XML files contain resource lists, as specified in [Resource List Syntax \(section 2.2.1\)](#). The TSWPP server generates these XML files and passes them back to the TSWPP client when the client requests them, so that each user can get a unique XML file. The files that are referenced by the XML file are stored on the server so that the client can download them on subsequent connections without the additional overhead of retrieving the resources.

3.1.1.3 Icon Files

Icon files represent icons and are stored on the TSWPP server file system. An XML resource list references icon files so that a TSWPP client can download and store them on the client machine for use in the presentation of the operating system user interface. <1>

3.1.1.4 Resource Files

Resource files represent the resources that are stored on the TSWPP server file system. An XML resource list references resource files so that a TSWPP client can download and store them on the client machine for use in launching remote applications from the operating system user interface. <2>

3.1.1.5 Resources

Resources are stored on the TSWPP server in memory and comprise the list of resources that are used to generate the [XML files \(section 3.1.1.2\)](#) that are sent to the TSWPP client. The data stored for a resource includes any [icon files \(section 3.1.1.3\)](#) associated with that resource as well as the [resource file \(section 3.1.1.4\)](#) itself. This data includes a friendly display name for the resource as well as a unique identifier. In addition, if the server implementation makes XML files unique for each user, then the information to determine whether a user is authorized to see a particular resource is also stored.

3.2 Client Details

3.2.1 Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. The described organization is provided to facilitate the explanation of how the protocol behaves. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that described in this document.

3.2.2 Timers

None.

3.2.3 Initialization

None.

3.2.4 Higher-Layer Triggered Events

None.

3.2.5 Message Processing Events and Sequencing Rules

The only sequence rule is that the client MUST initiate a request for an XML file (section [3.1.1.2](#)).

The client can request the resources that are described in the XML file in separate HTTP requests (section [2.1](#)), and the order of those requests is not significant.

3.2.6 Timer Events

None.

3.2.7 Other Local Events

None.

3.3 Server Details

3.3.1 Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. The described organization is provided to facilitate the explanation of how the protocol behaves. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that described in this document.

3.3.1.1 Authentication Cookie

When a TSWPP client first connects to the TSWPP server, an authentication cookie is transmitted to the client, as specified in section [3.1](#). This cookie contains information to authenticate a user to the server, as well as the security identifier (SID), as defined in [\[MS-DTYP\]](#) section 2.4.2, for the user that originally authenticated to the server.

3.3.2 Timers

None.

3.3.3 Initialization

None.

3.3.4 Higher-Layer Triggered Events

The TSWPP server SHOULD be implemented so that changing a resource triggers the update of an XML resource list that contains that resource. How this event is triggered is implementation-specific, and it depends on how the server obtains data about resources that are being included in XML resource lists.

3.3.5 Message Processing Events and Sequencing Rules

When a client makes a request for an XML file, the server MUST return the requested file. The set of resource elements contained in the [XML file \(section 3.1.1.2\)](#) are determined entirely by the identity of the user making the request. If the server is unable to provide the requested file to the client, the server MUST return an HTTP "404 Not Found" error code [\[RFC2616\]](#).

3.3.6 Timer Events

None.

3.3.7 Other Local Events

None.

4 Protocol Examples

4.1 Schema Version 1.1 Examples

4.1.1 Message with One Hosting Terminal Server

The following is an example of a message adhering to schema version 1.1 that specifies only one <HostingTerminalServer> element.

```
<?xml version="1.0" encoding="utf-8" ?>
<ResourceCollection PubDate="2008-08-09T20:49:22.0525806Z"
  SchemaVersion="1.1" xmlns="http://schemas.microsoft.com/ts/2007/05/tswf">
  <Publisher LastUpdated="2008-08-09T20:49:13.1463876Z" Name="TSWF Publisher"
    ID="C5DBDC2D-AED2-3871-5216-B4B26E523DAC" Description="A test publisher">
  <Resources>
  <Resource ID="fbe574600b3b5d1a7ac0a96713a35f8461b8bd92" Alias="calc"
    Title="Calculator" LastUpdated="2008-08-09T20:49:13.0995129Z"
    Type="RemoteApp" ExecutableName="calc.exe">
  <Icons>
  <IconRaw FileType="Ico" FileURL="/TS/images/calc032x32.ico" />
  <Icon32 Dimensions="32x32" FileType="Png"
    FileURL="/TS/images/calc032x32.png" />
  </Icons>
  <FileExtensions />
  <HostingTerminalServers>
  <HostingTerminalServer>
  <ResourceFile FileExtension=".rdp" URL="/TS/images/KL-TESTBED2-
    calc.rdp" />
  <TerminalServerRef Ref="KL-TESTBED2" />
  </HostingTerminalServer>
  </HostingTerminalServers>
  </Resource>
  <Resource ID="142b2423202d992dca556a47595a470de5b9d13b"
    Alias="CompMgmtLauncher" Title="Server Manager" LastUpdated="2008-08-
    09T20:49:13.0995129Z" Type="RemoteApp"
    ExecutableName="CompMgmtLauncher.exe">
  <Icons>
  <IconRaw FileType="Ico" FileURL="/TS/images/svrmgrnc032x32.ico" />
  <Icon32 Dimensions="32x32" FileType="Png"
    FileURL="/TS/images/svrmgrnc032x32.png" />
  </Icons>
  <FileExtensions />
  <HostingTerminalServers>
  <HostingTerminalServer>
  <ResourceFile FileExtension=".rdp" URL="/TS/images/KL-TESTBED2-
    CompMgmtLauncher.rdp" />
  <TerminalServerRef Ref="KL-TESTBED2" />
  </HostingTerminalServer>
  </HostingTerminalServers>
  </Resource>
  <Resource ID="befaf52a8a9f68461570872ab5fe1e22712b829e"
    Alias="mspaint (1)" Title="Paint" LastUpdated="2008-08-
    09T20:49:13.0995129Z" Type="RemoteApp" ExecutableName="mspaint.exe">
  <Icons>
  <IconRaw FileType="Ico" FileURL="/TS/images/mspaint032x32.ico" />
  <Icon32 Dimensions="32x32" FileType="Png"
    FileURL="/TS/images/mspaint032x32.png" />
  </Icons>
```

```

</Icons>
<FileExtensions />
<HostingTerminalServers>
  <HostingTerminalServer>
    <ResourceFile FileExtension=".rdp" URL="/TS/images/KL-TESTBED2-
      mspaint (1).rdp" />
    <TerminalServerRef Ref="KL-TESTBED2" />
  </HostingTerminalServer>
</HostingTerminalServers>
</Resource>
<Resource ID="a883f1456458daabfa96b82f9e127b400fced1b7" Alias="wordpad"
  Title="WordPad" LastUpdated="2008-08-09T20:49:13.0995129Z"
  Type="RemoteApp" ExecutableName="wordpad.exe">
<Icons>
  <IconRaw FileType="Ico" FileURL="/TS/images/wordpad032x32.ico" />
  <Icon32 Dimensions="32x32" FileType="Png"
    FileURL="/TS/images/wordpad032x32.png" />
</Icons>
<FileExtensions />
<HostingTerminalServers>
  <HostingTerminalServer>
    <ResourceFile FileExtension=".rdp" URL="/TS/images/KL-TESTBED2-
      wordpad.rdp" />
    <TerminalServerRef Ref="KL-TESTBED2" />
  </HostingTerminalServer>
</HostingTerminalServers>
</Resource>
<Resource ID="1d3d1c6ed34b7719dd6fa2893f6f1e78d983f844"
  Alias="TSRemoteDesktop" Title="Remote Desktop" LastUpdated="2008-08-
  09T20:49:13.1463876Z" Type="Desktop">
<Icons>
  <IconRaw FileType="Ico" FileURL="/TS/images/mstsc25632x32.ico" />
  <Icon32 Dimensions="32x32" FileType="Png"
    FileURL="/TS/images/mstsc25632x32.png" />
</Icons>
<FileExtensions />
<HostingTerminalServers>
  <HostingTerminalServer>
    <ResourceFile FileExtension=".rdp" URL="/TS/images/KL-TESTBED2-
      TSRemoteDesktop.rdp" />
    <TerminalServerRef Ref="KL-TESTBED2" />
  </HostingTerminalServer>
</HostingTerminalServers>
</Resource>
</Resources>
<TerminalServers>
  <TerminalServer ID="KL-TESTBED2" Name="KL-TESTBED2" LastUpdated="2008-08-
    09T20:49:13.1463876Z">
  </TerminalServer>
</TerminalServers>
</Publisher>
</ResourceCollection>

```

4.1.2 Message with Multiple Terminal Servers

The following is an example of a message adhering to schema version 1.1 that specifies multiple <HostingTerminalServer> elements.

```

<?xml version="1.0" encoding="utf-8" ?>
<ResourceCollection PubDate="2008-08-09T21:04:44.4216774Z"
  SchemaVersion="1.0" xmlns="http://schemas.microsoft.com/ts/2007/05/tswf">
<Publisher LastUpdated="2008-08-09T21:04:30.0467694Z" Name="TSWF Default
  Publisher" ID="C5DBDC2D-AED2-3871-5216-B4B26E523DAC" Description="A test
  publisher">
<Resources>
<Resource ID="6e4514834a0969aaa34b36149a6e5db453a4347e"
  Alias="wmplayer-kl-testbed4-VmFarm" Title="Windows Media Player"
  LastUpdated="2008-08-09T21:04:29.9842698Z" Type="RemoteApp">
<Icons>
  <IconRaw FileType="Ico" FileURL="/TS/images/wmplayer-kl-testbed4-
    VmFarm032x32.ico" />
  <Icon32 Dimensions="32x32" FileType="Png" FileURL="/TS/images/wmplayer-
    kl-testbed4-VmFarm032x32.png" />
</Icons>
<FileExtensions />
<HostingTerminalServers>
  <HostingTerminalServer>
    <ResourceFile FileExtension=".rdp" URL="/TS/images/KL-TESTBED4-wmplayer-
      kl-testbed4-VmFarm.rdp" />
    <TerminalServerRef Ref="KL-TESTBED4" />
  </HostingTerminalServer>
</HostingTerminalServers>
</Resource>
<Resource ID="95c33f5c2e9042e18e046c7fae8248fa91a8c399" Alias="dfrgui-kl-
  testbed4-VmFarm" Title="Disk Defragmenter" LastUpdated="2008-08-
  09T21:04:29.9842698Z" Type="RemoteApp">
<Icons>
  <IconRaw FileType="Ico" FileURL="/TS/images/dfrgui-kl-testbed4-
    VmFarm032x32.ico" />
  <Icon32 Dimensions="32x32" FileType="Png" FileURL="/TS/images/dfrgui-kl-
    testbed4-VmFarm032x32.png" />
</Icons>
<FileExtensions />
<HostingTerminalServers>
  <HostingTerminalServer>
    <ResourceFile FileExtension=".rdp" URL="/TS/images/KL-TESTBED4-dfrgui-
      kl-testbed4-VmFarm.rdp" />
    <TerminalServerRef Ref="KL-TESTBED4" />
  </HostingTerminalServer>
</HostingTerminalServers>
</Resource>
<Resource ID="e7692debcdbcabebfa3acfe0de2c84cfa05e5b18e" Alias="cleanmgr-kl-
  testbed4-VmFarm" Title="Disk Cleanup" LastUpdated="2008-08-
  09T21:04:29.9998947Z" Type="RemoteApp">
<Icons>
  <IconRaw FileType="Ico" FileURL="/TS/images/cleanmgr-kl-testbed4-
    VmFarm032x32.ico" />
  <Icon32 Dimensions="32x32" FileType="Png" FileURL="/TS/images/cleanmgr-
    kl-testbed4-VmFarm032x32.png" />
</Icons>
<FileExtensions />
<HostingTerminalServers>
  <HostingTerminalServer>
    <ResourceFile FileExtension=".rdp" URL="/TS/images/KL-TESTBED4-cleanmgr-
      kl-testbed4-VmFarm.rdp" />
    <TerminalServerRef Ref="KL-TESTBED4" />
  </HostingTerminalServer>
</HostingTerminalServers>

```



```

</HostingTerminalServers>
</Resource>
<Resource ID="aa8cfd93a7028f13b1c6e0579923f824ea9f0cbf"
Alias="TSRemoteDesktop-localhost-VmFarm" Title="Remote Desktop"
LastUpdated="2008-08-09T21:04:30.0155196Z" Type="RemoteApp">
<Icons>
<IconRaw FileType="Ico" FileURL="/TS/images/TSRemoteDesktop-localhost-
VmFarm32x32.ico" />
<Icon32 Dimensions="32x32" FileType="Png"
FileURL="/TS/images/mstsc25632x32.png" />
</Icons>
<FileExtensions />
<HostingTerminalServers>
<HostingTerminalServer>
<ResourceFile FileExtension=".rdp" URL="/TS/images/KL-TESTBED2-
TSRemoteDesktop-localhost-VmFarm.rdp" />
<TerminalServerRef Ref="KL-TESTBED2" />
</HostingTerminalServer>
</HostingTerminalServers>
</Resource>
<Resource ID="ed04af6be1cc8fe836bbb136dbe61f07162551e9" Alias="wordpad-
localhost-VmFarm" Title="WordPad" LastUpdated="2008-08-
09T21:04:30.0155196Z" Type="RemoteApp">
<Icons>
<IconRaw FileType="Ico" FileURL="/TS/images/wordpad-localhost-
VmFarm032x32.ico" />
<Icon32 Dimensions="32x32" FileType="Png" FileURL="/TS/images/wordpad-
localhost-VmFarm032x32.png" />
</Icons>
<FileExtensions />
<HostingTerminalServers>
<HostingTerminalServer>
<ResourceFile FileExtension=".rdp" URL="/TS/images/KL-TESTBED2-wordpad-
localhost-VmFarm.rdp" />
<TerminalServerRef Ref="KL-TESTBED2" />
</HostingTerminalServer>
</HostingTerminalServers>
</Resource>
<Resource ID="5c97c0e4e29031e123965236f9d0385f863faa9a" Alias="mspaint
(1)-localhost-VmFarm" Title="Paint" LastUpdated="2008-08-
09T21:04:30.0311445Z" Type="RemoteApp">
<Icons>
<IconRaw FileType="Ico" FileURL="/TS/images/mspaint__1_-localhost-
VmFarm032x32.ico" />
<Icon32 Dimensions="32x32" FileType="Png"
FileURL="/TS/images/mspaint__1_-localhost-VmFarm032x32.png" />
</Icons>
<FileExtensions />
<HostingTerminalServers>
<HostingTerminalServer>
<ResourceFile FileExtension=".rdp" URL="/TS/images/KL-TESTBED2-
mspaint__1_-localhost-VmFarm.rdp" />
<TerminalServerRef Ref="KL-TESTBED2" />
</HostingTerminalServer>
</HostingTerminalServers>
</Resource>
<Resource ID="34cf1eb036041f5e31c20cb569e9125b8fdc4edb"
Alias="CompMgmtLauncher-localhost-VmFarm" Title="Server Manager"
LastUpdated="2008-08-09T21:04:30.0467694Z" Type="RemoteApp">

```

```

<Icons>
  <IconRaw FileType="Ico" FileURL="/TS/images/CompMgmtLauncher-localhost-
    VmFarm032x32.ico" />
  <Icon32 Dimensions="32x32" FileType="Png"
    FileURL="/TS/images/CompMgmtLauncher-localhost-VmFarm032x32.png" />
</Icons>
<FileExtensions />
<HostingTerminalServers>
  <HostingTerminalServer>
    <ResourceFile FileExtension=".rdp" URL="/TS/images/KL-TESTBED2-
      CompMgmtLauncher-localhost-VmFarm.rdp" />
    <TerminalServerRef Ref="KL-TESTBED2" />
  </HostingTerminalServer>
</HostingTerminalServers>
</Resource>
<Resource ID="35a43a391cb9654cfd2c5b6d732fd054667e2fc6" Alias="calc-
  localhost-VmFarm" Title="Calculator" LastUpdated="2008-08-
  09T21:04:30.0467694Z" Type="RemoteApp">
  <Icons>
    <IconRaw FileType="Ico" FileURL="/TS/images/calc-localhost-
      VmFarm032x32.ico" />
    <Icon32 Dimensions="32x32" FileType="Png" FileURL="/TS/images/calc-
      localhost-VmFarm032x32.png" />
  </Icons>
  <FileExtensions />
  <HostingTerminalServers>
    <HostingTerminalServer>
      <ResourceFile FileExtension=".rdp" URL="/TS/images/KL-TESTBED2-calc-
        localhost-VmFarm.rdp" />
      <TerminalServerRef Ref="KL-TESTBED2" />
    </HostingTerminalServer>
  </HostingTerminalServers>
</Resource>
</Resources>
<TerminalServers>
  <TerminalServer ID="KL-TESTBED2" Name="KL-TESTBED2" LastUpdated="2008-08-
    09T21:04:30.0467694Z">
  </TerminalServer>
  <TerminalServer ID="KL-TESTBED4" Name="KL-TESTBED4" LastUpdated="2008-08-
    09T21:04:30.0467694Z">
  </TerminalServer>
</TerminalServers>
</Publisher>
</ResourceCollection>

```

4.2 .ASPXAUTH Cookie Message Returned from the Server

The following is an example of the **.ASPXAUTH** cookie that is returned from the server after the server has authorized the user for access. The contents of the cookie will then be set as a Cookie, in the client's responses, with the name of **.ASPXAUTH**.

```

HTTP/1.1 200 OK
Cache-Control: private
Content-Type: application/x-msts-webfeed-login; charset=utf-8
Server: Microsoft-IIS/7.5
X-AspNet-Version: 2.0.50727
Persistent-Auth: true

```

X-Powered-By: ASP.NET
Date: Thu, 11 Mar 2010 00:51:13 GMT
Content-Length: 384

Note Contents of server-generated cookie are as defined in section [2.2.3](#)

5 Security

5.1 Security Considerations for Implementers

None.

5.2 Index of Security Parameters

None.

6 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Windows® 7 operating system
- Windows Server® 2008 R2 operating system

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

[<1> Section 3.1.1.3](#): Windows client implementations use the information in icon files in the presentation of the Windows **Start** menu.

[<2> Section 3.1.1.4](#): Windows client implementations use the information in resource files in launching remote applications from the Windows **Start** menu.

7 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

8 Index

A

Abstract data model
[client](#) 19
[server](#) 19
[Applicability](#) 7

C

[Capability negotiation](#) 7
[Change tracking](#) 30
Client
[abstract data model](#) 19
[higher-layer triggered events](#) 20
[initialization](#) 20
[local events](#) 21
[message processing](#) 20
[overview](#) 17
[sequencing rules](#) 20
[timer events](#) 21
[timers](#) 20

D

Data model - abstract
[client](#) 19
[server](#) 19

E

Examples
[message with multiple terminal servers and gateways](#) 23
[message with no gateway servers and one hosting terminal server](#) 22

F

[Fields - vendor-extensible](#) 8

G

[Glossary](#) 5

H

Higher-layer triggered events
[client](#) 20
[server](#) 20

I

[Implementer - security considerations](#) 28
[Index of security parameters](#) 28
[Informative references](#) 6
Initialization
[client](#) 20
[server](#) 20
[Introduction](#) 5

L

Local events
[client](#) 21
[server](#) 21

M

Message processing
[client](#) 20
[server](#) 20
[Message with multiple terminal servers and gateways example](#) 23
[Message with no gateway servers and one hosting terminal server example](#) 22
Messages
[syntax](#) 9
[transport](#) 9

N

[Normative references](#) 6

O

[Overview \(synopsis\)](#) 7

P

[Parameters - security index](#) 28
[Preconditions](#) 7
[Prerequisites](#) 7
[Product behavior](#) 29

R

References
[informative](#) 6
[normative](#) 6
[Relationship to other protocols](#) 7

S

Security
[implementer considerations](#) 28
[parameter index](#) 28
Sequencing rules
[client](#) 20
[server](#) 20
Server
[abstract data model](#) 19
[higher-layer triggered events](#) 20
[initialization](#) 20
[local events](#) 21
[message processing](#) 20
[overview](#) 17
[sequencing rules](#) 20
[timer events](#) 21

- [timers](#) 20
- [Standards assignments](#) 8
- Syntax
 - [ResourceList syntax](#) 9
 - schema element definitions
 - [FileExtension](#) 15
 - [FileExtensions](#) 15
 - [HostingTerminalServer](#) 14
 - [HostingTerminalServers](#) 14
 - [Icon](#) 13
 - [Icons](#) 13
 - [overview](#) 12
 - [Publisher](#) 12
 - [Resource](#) 13
 - [ResourceCollection](#) 12
 - [ResourceFile](#) 14
 - [Resources](#) 13
 - [TerminalServer](#) 14
 - [TerminalServerRef](#) 14
 - [TerminalServers](#) 14

T

- Timer events
 - [client](#) 21
 - [server](#) 21
- Timers
 - [client](#) 20
 - [server](#) 20
- [Tracking changes](#) 30
- [Transport](#) 9
- Triggered events - higher-layer
 - [client](#) 20
 - [server](#) 20

V

- [Vendor-extensible fields](#) 8
- [Versioning](#) 7