[MS-RASA]:
Remote Access Server Advertisement (RASADV) 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 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.
- **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
07/20/2007	0.1	Major	MCPP Milestone 5 Initial Availability
09/28/2007	0.1.1	Editorial	Revised and edited the technical content.
10/23/2007	0.1.2	Editorial	Revised and edited the technical content.
11/30/2007	0.1.3	Editorial	Revised and edited the technical content.
01/25/2008	0.1.4	Editorial	Revised and edited the technical content.
03/14/2008	0.1.5	Editorial	Revised and edited the technical content.
05/16/2008	0.1.6	Editorial	Revised and edited the technical content.
06/20/2008	1.0	Major	Updated and revised the technical content.
07/25/2008	1.0.1	Editorial	Revised and edited the technical content.
08/29/2008	2.0	Major	Added examples.
10/24/2008	3.0	Major	Updated and revised the technical content.
12/05/2008	3.1	Minor	Updated the technical content.
01/16/2009	3.1.1	Editorial	Revised and edited the technical content.
02/27/2009	3.1.2	Editorial	Revised and edited the technical content.
04/10/2009	3.1.3	Editorial	Revised and edited the technical content.
05/22/2009	3.1.4	Editorial	Revised and edited the technical content.
07/02/2009	3.2	Minor	Updated the technical content.
08/14/2009	3.2.1	Editorial	Revised and edited the technical content.
09/25/2009	3.3	Minor	Updated the technical content.
11/06/2009	3.3.1	Editorial	Revised and edited the technical content.
12/18/2009	3.3.2	Editorial	Revised and edited the technical content.
01/29/2010	3.3.3	Editorial	Revised and edited the technical content.
03/12/2010	3.3.4	Editorial	Revised and edited the technical content.
04/23/2010	3.3.5	Editorial	Revised and edited the technical content.
06/04/2010	3.3.6	Editorial	Revised and edited the technical content.
07/16/2010	3.3.6	No change	No changes to the meaning, language, or formatting of the technical content.

Date	Revision History	Revision Class	Comments
08/27/2010	3.3.6	No change	No changes to the meaning, language, or formatting of the technical content.
10/08/2010	3.3.6	No change	No changes to the meaning, language, or formatting of the technical content.
11/19/2010	3.3.6	No change	No changes to the meaning, language, or formatting of the technical content.
01/07/2011	3.3.6	No change	No changes to the meaning, language, or formatting of the technical content.
02/11/2011	3.3.6	No change	No changes to the meaning, language, or formatting of the technical content.
03/25/2011	3.3.6	No change	No changes to the meaning, language, or formatting of the technical content.
05/06/2011	3.3.6	No change	No changes to the meaning, language, or formatting of the technical content.
06/17/2011	3.4	Minor	Clarified the meaning of the technical content.

Contents

1 Introduction

This document specifies the Remote Access Server Advertisement (RASADV) Protocol, by which **Remote Access Service (RAS) servers** advertise their presence within a local network, which allows network administrators to detect nonmalicious configuration and deployment of gateways providing external access to their network.

1.1 Glossary

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

ASCII
domain
domain name (3)
Remote Access Service (RAS) server

The following terms are specific to this document:

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

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.

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.

[IANAIMA] Internet Assigned Numbers Authority, "Internet Multicast Addresses", March 2007, http://www.iana.org/assignments/multicast-addresses

[IANAPORT] Internet Assigned Numbers Authority, "Port Numbers", November 2006, http://www.iana.org/assignments/port-numbers

[RFC768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, August 1980, http://www.ietf.org/rfc/rfc768.txt

[RFC791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981, http://www.ietf.org/rfc/rfc791.txt

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

1.2.2 Informative References

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

1.3 Overview

The RASADV Protocol is a simple, unidirectional, multicast protocol. The sender is a RAS server, which puts its machine name in a message and periodically multicasts it on its local network.<1> The receiver listens for periodic messages and passes the message content (the sender's machine name) and the source address to an application.<2> A typical application using the listener side of this protocol displays the information to the user.

1.4 Relationship to Other Protocols

The RASADV Protocol depends on the **User Datagram Protocol (UDP)**, as specified in [RFC768], as a transport. No other protocols depend on the RASADV Protocol.

1.5 Prerequisites/Preconditions

1.6 Applicability Statement

The RASADV Protocol applies only to detecting intentional or accidental configuration and deployment of servers over time. It does not apply to detecting malicious configuration and deployment. It also does not apply if an administrator requires the ability to quickly detect such servers on demand, because it relies on a periodic broadcast mechanism.

1.7 Versioning and Capability Negotiation

The RASADV Protocol has no versioning or capability negotiation capabilities.

1.8 Vendor-Extensible Fields

This protocol has no vendor-extensible fields.

1.9 Standards Assignments

The RASADV Protocol uses the following standards assignments.

Parameter	Value	Reference
UDP port number	9753	[IANAPORT]
Internet Protocol version 4 (IPv4) multicast address	239.255.2.2	[IANAIMA]

2 Messages

The following sections specify how RASADV Protocol messages are transported and common RASADV Protocol data types.

2.1 Transport

All messages MUST be sent over UDP, as specified in [RFC768], with the UDP destination port set to 9753 and the IP destination address set to 239.255.2.2. The IP Time to Live (TTL), as specified in [RFC791] section 3.1, SHOULD be set to 15.

2.2 Message Syntax

The RASADV Protocol has a single message type.

2.2.1 Server Advertisement

The UDP message MUST be one of the following:

1. The **ASCII** string "Hostname=<hostname>", where *<hostname>* is replaced with the host name of the sender, followed by a line feed (0x0A) and a null character (0x00). This syntax MUST be used if the sender is not a member of a **domain**.

2. The ASCII string "Hostname=<hostname>\nDomain=<domainname>", where *<hostname>* is replaced with the host name of the sender, followed by a line feed (0x0A), and *<domainname>* is replaced with the **domain name** of the domain of which the machine is a member, followed by a line feed (0x0A) and a null character (0x00). This syntax MUST be used if the sender is a member of a domain.

3 Protocol Details

The following sections specify details of the RASADV Protocol, including abstract data models and message processing rules.

3.1 RAS Server Details

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.

Host Name: The host name of the RAS server.

3.1.2 Timers

Advertisement Timer: A periodic timer used to multicast a server advertisement. It SHOULD have a period of 1 hour.

3.1.3 Initialization

When a RAS server starts, it MUST immediately send a server advertisement, as specified in section 2.2.1, and start its Advertisement Timer.

3.1.4 Higher-Layer Triggered Events

This protocol has no higher-layer triggered events.

3.1.5 Message Processing Events and Sequencing Rules

The RAS server role has no message processing events or sequencing rules.

3.1.6 Timer Events

When the Advertisement Timer expires, the RAS server MUST send a server advertisement, as specified in section 2.2.1, and restart its Advertisement Timer.

3.1.7 Other Local Events

This protocol has no other local events.

3.2 Listener Details

3.2.1 Abstract Data Model

The listener role has no abstract data model.

3.2.2 Timers

The listener role has no timers.

3.2.3 Initialization

When a listener starts, it MUST start listening for UDP messages on port 9753 and join the IPv4 multicast group 239.255.2.2.

3.2.4 Higher-Layer Triggered Events

This protocol has no higher-layer triggered events.

3.2.5 Message Processing Events and Sequencing Rules

When a message arrives, the listener MUST deliver the message text and the source IP address to the application.

3.2.6 Timer Events

The listener role has no timer events.

3.2.7 Other Local Events

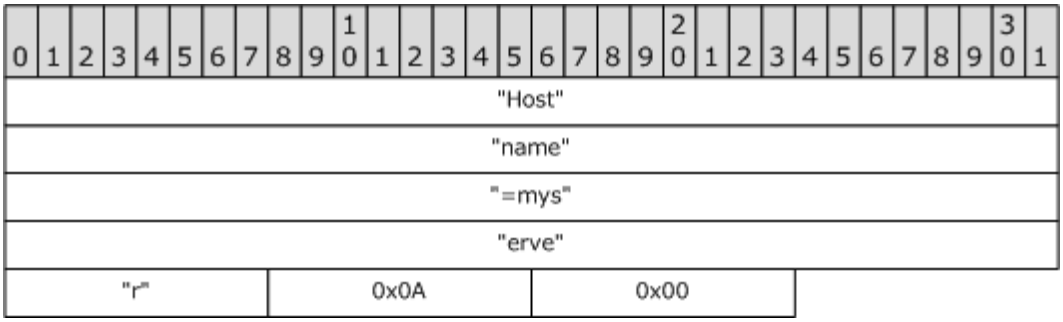
This protocol has no other local events.

4 Protocol Examples

Following are two examples where a RAS server broadcasts its advertisement on the local network.

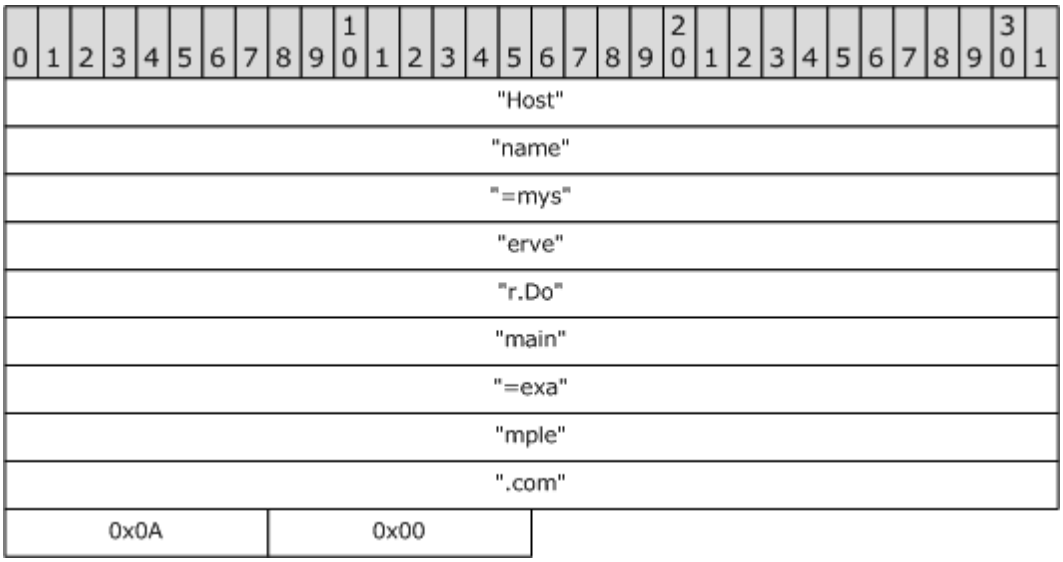
4.1 Example 1

The network administrator starts a listener tool. The tool begins listening on the RASADV port and the multicast address. Later, the user configures a RAS server on the local network, on a computer named "myserver", which is not a member of a domain. When the RAS server starts, and every hour thereafter, it sends a Server Advertisement message containing the string "Hostname=myserver" followed by a line feed (0x0A) and a null character (0x00).



4.2 Example 2

The network administrator starts a listener tool. The tool begins listening on the RASADV port and the multicast address. Later, the user configures a RAS server on the local network, on a computer named "myserver", which is a member of the domain named "example.com". When the RAS server starts, and every hour thereafter, it sends a Server Advertisement message containing the string "Hostname=myserver.Domain=example.com" followed by a line feed (0x0A) and a null character (0x00).



4.3 Examples Sequence Diagram

The following sequence diagram illustrates the UDP messages corresponding to the examples mentioned previously.

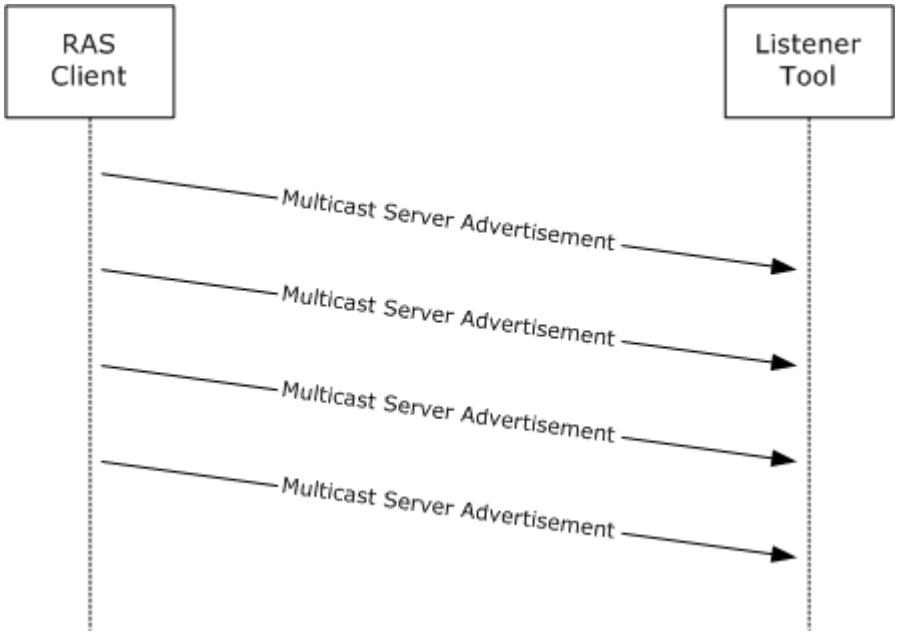


Figure 1: RAS server successfully multicasting its advertisement RAS client

5 Security

The following sections specify security considerations for implementers of the RASADV Protocol.

5.1 Security Considerations for Implementers

RASADV assumes that servers advertise themselves in order to be detected, and therefore does not provide any security or the ability to detect malicious servers.

5.2 Index of Security Parameters

This protocol has no security parameters.

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:

- Microsoft 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

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 1.3: Windows 2000 Server, Windows Server 2003, Windows Server 2008, and Windows Server 2008 R2 support the sender role in Routing and Remote Access, which is an add-on that combines remote access support with network routing functionality.

<2> Section 1.3: The listener is implemented in Windows by the command "netsh ras show activeservers", which displays the messages as they arrive.

7 Change Tracking

This section identifies changes that were made to the [MS-RASA] protocol document between the May 2011 and June 2011 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

The revision class **New** means that a new document is being released.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.
- An extensive rewrite, addition, or deletion of major portions of content.
- The removal of a document from the documentation set.
- Changes made for template compliance.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **Editorial** means that the language and formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical or language changes were introduced. The technical content of the document is identical to the last released version, but minor editorial and formatting changes, as well as updates to the header and footer information, and to the revision summary, may have been made.

Major and minor changes can be described further using the following change types:

- New content added.
- Content updated.
- Content removed.
- New product behavior note added.
- Product behavior note updated.
- Product behavior note removed.
- New protocol syntax added.
- Protocol syntax updated.
- Protocol syntax removed.
- New content added due to protocol revision.
- Content updated due to protocol revision.
- Content removed due to protocol revision.
- New protocol syntax added due to protocol revision.

- Protocol syntax updated due to protocol revision.
- Protocol syntax removed due to protocol revision.
- New content added for template compliance.
- Content updated for template compliance.
- Content removed for template compliance.
- Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated.**

Some important terms used in the change type descriptions are defined as follows:

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.
- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

The changes made to this document are listed in the following table. For more information, please contact protocol@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
1.2 References	Added explanatory statement regarding the removal of the publishing year from Microsoft Open Specification document references.	N	Content updated.

8 Index

V