

## [MS-RASA]:

# Remote Access Server Advertisement (RASADV) Protocol

---

### 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](mailto: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](http://www.microsoft.com/trademarks).
- **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
7/20/2007	0.1	Major	MCPP Milestone 5 Initial Availability
9/28/2007	0.1.1	Editorial	Changed language and formatting in the technical content.
10/23/2007	0.1.2	Editorial	Changed language and formatting in the technical content.
11/30/2007	0.1.3	Editorial	Changed language and formatting in the technical content.
1/25/2008	0.1.4	Editorial	Changed language and formatting in the technical content.
3/14/2008	0.1.5	Editorial	Changed language and formatting in the technical content.
5/16/2008	0.1.6	Editorial	Changed language and formatting in the technical content.
6/20/2008	1.0	Major	Updated and revised the technical content.
7/25/2008	1.0.1	Editorial	Changed language and formatting in the technical content.
8/29/2008	2.0	Major	Added examples.
10/24/2008	3.0	Major	Updated and revised the technical content.
12/5/2008	3.1	Minor	Clarified the meaning of the technical content.
1/16/2009	3.1.1	Editorial	Changed language and formatting in the technical content.
2/27/2009	3.1.2	Editorial	Changed language and formatting in the technical content.
4/10/2009	3.1.3	Editorial	Changed language and formatting in the technical content.
5/22/2009	3.1.4	Editorial	Changed language and formatting in the technical content.
7/2/2009	3.2	Minor	Clarified the meaning of the technical content.
8/14/2009	3.2.1	Editorial	Changed language and formatting in the technical content.
9/25/2009	3.3	Minor	Clarified the meaning of the technical content.
11/6/2009	3.3.1	Editorial	Changed language and formatting in the technical content.
12/18/2009	3.3.2	Editorial	Changed language and formatting in the technical content.
1/29/2010	3.3.3	Editorial	Changed language and formatting in the technical content.
3/12/2010	3.3.4	Editorial	Changed language and formatting in the technical content.
4/23/2010	3.3.5	Editorial	Changed language and formatting in the technical content.
6/4/2010	3.3.6	Editorial	Changed language and formatting in the technical content.
7/16/2010	3.3.6	None	No changes to the meaning, language, or formatting of the technical content.
8/27/2010	3.3.6	None	No changes to the meaning, language, or formatting of the technical content.
10/8/2010	3.3.6	None	No changes to the meaning, language, or formatting of the technical content.
11/19/2010	3.3.6	None	No changes to the meaning, language, or formatting of the

<b>Date</b>	<b>Revision History</b>	<b>Revision Class</b>	<b>Comments</b>
			technical content.
1/7/2011	3.3.6	None	No changes to the meaning, language, or formatting of the technical content.
2/11/2011	3.3.6	None	No changes to the meaning, language, or formatting of the technical content.
3/25/2011	3.3.6	None	No changes to the meaning, language, or formatting of the technical content.
5/6/2011	3.3.6	None	No changes to the meaning, language, or formatting of the technical content.
6/17/2011	3.4	Minor	Clarified the meaning of the technical content.
9/23/2011	3.4	None	No changes to the meaning, language, or formatting of the technical content.
12/16/2011	4.0	Major	Updated and revised the technical content.
3/30/2012	4.0	None	No changes to the meaning, language, or formatting of the technical content.
7/12/2012	4.0	None	No changes to the meaning, language, or formatting of the technical content.
10/25/2012	4.0	None	No changes to the meaning, language, or formatting of the technical content.
1/31/2013	4.0	None	No changes to the meaning, language, or formatting of the technical content.
8/8/2013	5.0	Major	Updated and revised the technical content.
11/14/2013	5.0	None	No changes to the meaning, language, or formatting of the technical content.
2/13/2014	5.0	None	No changes to the meaning, language, or formatting of the technical content.
5/15/2014	5.0	None	No changes to the meaning, language, or formatting of the technical content.
6/30/2015	6.0	Major	Significantly changed the technical content.
10/16/2015	6.0	No Change	No changes to the meaning, language, or formatting of the technical content.

# Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>5</b>
1.1	Glossary .....	5
1.2	References .....	5
1.2.1	Normative References .....	5
1.2.2	Informative References .....	6
1.3	Overview .....	6
1.4	Relationship to Other Protocols .....	6
1.5	Prerequisites/Preconditions .....	6
1.6	Applicability Statement .....	6
1.7	Versioning and Capability Negotiation .....	6
1.8	Vendor-Extensible Fields .....	6
1.9	Standards Assignments.....	6
<b>2</b>	<b>Messages.....</b>	<b>8</b>
2.1	Transport .....	8
2.2	Message Syntax.....	8
2.2.1	Server Advertisement .....	8
<b>3</b>	<b>Protocol Details.....</b>	<b>9</b>
3.1	RAS Server Details .....	9
3.1.1	Abstract Data Model.....	9
3.1.2	Timers .....	9
3.1.3	Initialization.....	9
3.1.4	Higher-Layer Triggered Events .....	9
3.1.5	Message Processing Events and Sequencing Rules .....	9
3.1.6	Timer Events.....	9
3.1.7	Other Local Events.....	9
3.2	Listener Details.....	9
3.2.1	Abstract Data Model.....	9
3.2.2	Timers .....	9
3.2.3	Initialization.....	10
3.2.4	Higher-Layer Triggered Events .....	10
3.2.5	Message Processing Events and Sequencing Rules .....	10
3.2.6	Timer Events.....	10
3.2.7	Other Local Events.....	10
<b>4</b>	<b>Protocol Examples.....</b>	<b>11</b>
4.1	Example 1.....	11
4.2	Example 2.....	11
4.3	Examples Sequence Diagram.....	11
<b>5</b>	<b>Security.....</b>	<b>13</b>
5.1	Security Considerations for Implementers .....	13
5.2	Index of Security Parameters .....	13
<b>6</b>	<b>Appendix A: Product Behavior .....</b>	<b>14</b>
<b>7</b>	<b>Change Tracking.....</b>	<b>15</b>
<b>8</b>	<b>Index.....</b>	<b>16</b>

# 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.

Sections 1.8, 2, and 3 of this specification are normative and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in [\[RFC2119\]](#). Sections 1.5 and 1.9 are also normative but do not contain those terms. All other sections and examples in this specification are informative.

## 1.1 Glossary

The following terms are specific to this document:

**ASCII:** The American Standard Code for Information Interchange (ASCII) is an 8-bit character-encoding scheme based on the English alphabet. ASCII codes represent text in computers, communications equipment, and other devices that work with text. ASCII refers to a single 8-bit ASCII character or an array of 8-bit ASCII characters with the high bit of each character set to zero.

**domain:** A set of users and computers sharing a common namespace and management infrastructure. At least one computer member of the set must act as a domain controller (DC) and host a member list that identifies all members of the domain, as well as optionally hosting the Active Directory service. The domain controller provides authentication (2) of members, creating a unit of trust for its members. Each domain has an identifier that is shared among its members. For more information, see [\[MS-AUTHSOD\]](#) section 1.1.1.5 and [\[MS-ADTS\]](#).

**domain name:** The name given by an administrator to a collection of networked computers that share a common directory. Part of the domain naming service naming structure, domain names consist of a sequence of name labels separated by periods.

**Remote Access Service (RAS) server:** A type of network access server (NAS) that provides modem dial-up or virtual private network (VPN) access to a network.

**User Datagram Protocol (UDP):** The connectionless protocol within TCP/IP that corresponds to the transport layer in the ISO/OSI reference model.

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

## 1.2 References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the [Errata](#).

### 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](mailto:dochelp@microsoft.com). We will assist you in finding the relevant information.

[IANAIMA] IANA, "Internet Multicast Addresses", March 2007, <http://www.iana.org/assignments/multicast-addresses>

[IANAPORT] IANA, "Service Name and Transport Protocol Port Number Registry", November 2006, <http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml>

[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>

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

[RFC791] Postel, J., Ed., "Internet Protocol: DARPA Internet Program Protocol Specification", RFC 791, September 1981, <http://www.rfc-editor.org/rfc/rfc791.txt>

## 1.2.2 Informative References

None.

## 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	<a href="#">[IANAPORT]</a>
Internet Protocol version 4 (IPv4) multicast address	239.255.2.2	<a href="#">[IANAIMA]</a>



## 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).

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
"Host"																															
"name"																															
"=mys"																															
"erve"																															
"r"								0x0A								0x00															

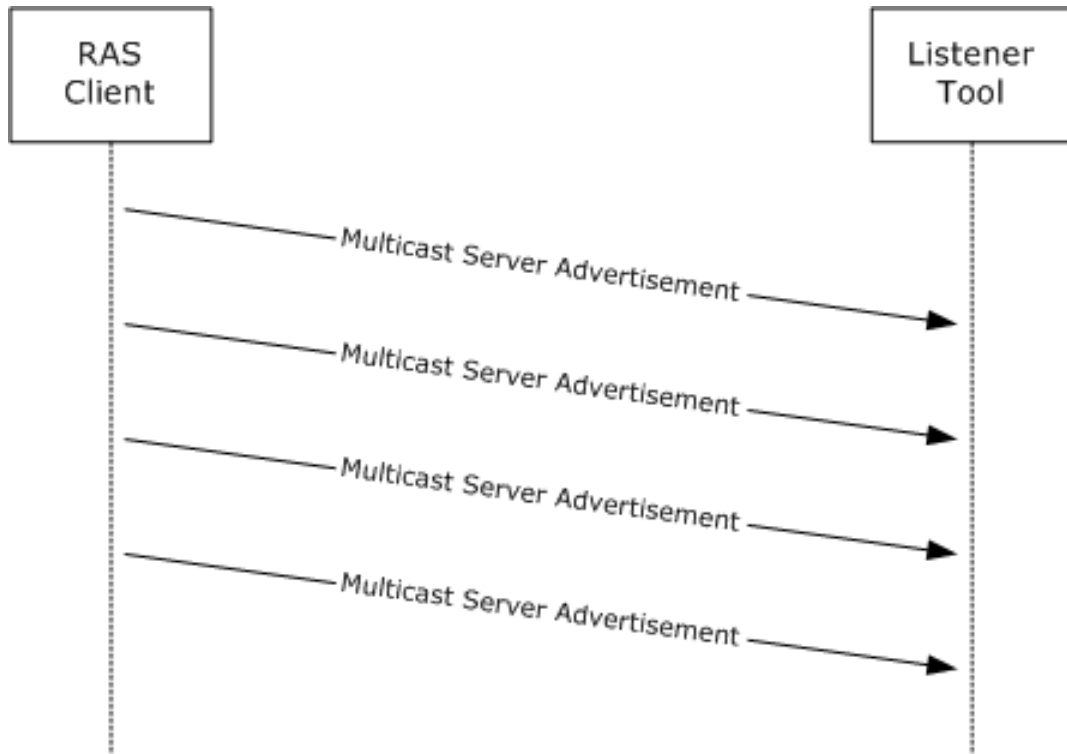
### 4.2 Example 2

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).

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
"Host"																															
"name"																															
"=mys"																															
"erve"																															
"r.Do"																															
"main"																															
"=exa"																															
"mple"																															
".com"																															
0x0A								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.

Note: Some of the information in this section is subject to change because it applies to a preliminary product version, and thus may differ from the final version of the software when released. All behavior notes that pertain to the preliminary product version contain specific references to it as an aid to the reader.

- 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
- Windows Server 2012 R2 operating system
- Windows 10 operating system
- Windows Server 2016 Technical Preview 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 operating system, Windows Server 2003, Windows Server 2008, Windows Server 2008 R2 operating system, Windows Server 2012, and Windows Server 2012 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

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  
[listener](#) 9  
[server](#) 9  
[Applicability](#) 6

### C

[Capability negotiation](#) 6  
[Change tracking](#) 15

### D

Data model - abstract  
[listener](#) 9  
[server](#) 9

### E

[Examples](#) 11

### F

[Fields - vendor-extensible](#) 6

### G

[Glossary](#) 5

### H

Higher-layer triggered events  
[listener](#) 10  
[server](#) 9

### I

[Implementer - security considerations](#) 13  
[Index of security parameters](#) 13  
[Informative references](#) 6  
Initialization  
[listener](#) 10  
[server](#) 9  
[Introduction](#) 5

### L

Listener  
[abstract data model](#) 9  
[higher-layer triggered events](#) 10  
[initialization](#) 10  
[local events](#) 10  
[message processing](#) 10  
[sequencing rules](#) 10  
[timer events](#) 10  
[timers](#) 9  
Local events  
[listener](#) 10  
[server](#) 9

### M

Message processing  
[listener](#) 10  
[server](#) 9  
Messages  
[overview](#) 8  
[Server Advertisement](#) 8  
[syntax](#) 8  
[transport](#) 8

### N

[Normative references](#) 5

### O

Other local events  
[server](#) 9  
[Overview](#) 6  
[Overview \(synopsis\)](#) 6

### P

[Parameters - security index](#) 13  
[Preconditions](#) 6  
[Prerequisites](#) 6  
[Product behavior](#) 14  
Protocol Details  
[overview](#) 9

### R

[References](#) 5  
[informative](#) 6  
[normative](#) 5  
[Relationship to other protocols](#) 6

### S

Security  
[implementer considerations](#) 13  
[overview](#) 13  
[parameter index](#) 13  
Sequencing rules  
[listener](#) 10  
[server](#) 9  
Server  
[abstract data model](#) 9  
[higher-layer triggered events](#) 9  
[initialization](#) 9  
[local events](#) 9  
[message processing](#) 9  
[other local events](#) 9  
[sequencing rules](#) 9  
[timer events](#) 9  
[timers](#) 9  
[Server advertisement](#) 8  
[Server Advertisement message](#) 8  
[Standards assignments](#) 6  
[Syntax](#) 8



## T

Timer events

[listener](#) 10

[server](#) 9

Timers

[listener](#) 9

[server](#) 9

[Tracking changes](#) 15

[Transport](#) 8

Triggered events - higher-layer

[listener](#) 10

[server](#) 9

## V

[Vendor-extensible fields](#) 6

[Versioning](#) 6