



Document Identifier: DSP1069

Date: 2016-02-23

Version: 1.0.0g

1  
2  
3  
4  
5

## 6 DNS Service Management Profile

### Information for Work-in-Progress version:

**IMPORTANT:** This document is not a standard. It does not necessarily reflect the views of the DMTF or its members. Because this document is a Work in Progress, this document may still change, perhaps profoundly and without notice. This document is available for public review and comment until superseded.

**Provide any comments through the DMTF Feedback Portal:**

<http://www.dmtf.org/standards/feedback>

7 **Supersedes: 1.0.0f**  
8 **Document Class: Normative**  
9 **Document Status: Work in Progress**  
10 **Document Language: en-US**

11

12 Copyright Notice

13 Copyright © 2014-2016 Distributed Management Task Force, Inc. (DMTF). All rights reserved.

14 DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems  
15 management and interoperability. Members and non-members may reproduce DMTF specifications and  
16 documents, provided that correct attribution is given. As DMTF specifications may be revised from time to  
17 time, the particular version and release date should always be noted.

18 Implementation of certain elements of this standard or proposed standard may be subject to third party  
19 patent rights, including provisional patent rights (herein "patent rights"). DMTF makes no representations  
20 to users of the standard as to the existence of such rights, and is not responsible to recognize, disclose,  
21 or identify any or all such third party patent right, owners or claimants, nor for any incomplete or  
22 inaccurate identification or disclosure of such rights, owners or claimants. DMTF shall have no liability to  
23 any party, in any manner or circumstance, under any legal theory whatsoever, for failure to recognize,  
24 disclose, or identify any such third party patent rights, or for such party's reliance on the standard or  
25 incorporation thereof in its product, protocols or testing procedures. DMTF shall have no liability to any  
26 party implementing such standard, whether such implementation is foreseeable or not, nor to any patent  
27 owner or claimant, and shall have no liability or responsibility for costs or losses incurred if a standard is  
28 withdrawn or modified after publication, and shall be indemnified and held harmless by any party  
29 implementing the standard from any and all claims of infringement by a patent owner for such  
30 implementations.

31 For information about patents held by third-parties which have notified the DMTF that, in their opinion,  
32 such patent may relate to or impact implementations of DMTF standards, visit  
33 <http://www.dmtf.org/about/policies/disclosures.php>.

34 This document's normative language is English. Translation into other languages is permitted.

# CONTENTS

36	Foreword .....	7
37	Introduction.....	8
38	Document conventions.....	8
39	1    Scope .....	9
40	2    Normative references.....	9
41	3    Terms and definitions .....	10
42	4    Symbols and abbreviated terms.....	11
43	5    Synopsis .....	12
44	6    Description .....	13
45	6.1    Class diagram .....	13
46	6.1.1    Support for IP protocol versioning .....	15
47	6.1.2    Management of DNS security .....	15
48	6.1.3    Representation of DNS service usage data .....	15
49	7    Implementation.....	15
50	7.1    Representing a DNS service .....	15
51	7.1.1    CIM_ProtocolService.Protocol.....	15
52	7.1.2    DNS service capabilities .....	15
53	7.1.3    Managing the DNS service's state.....	15
54	7.2    DNS service port management.....	16
55	7.2.1    UDP port .....	16
56	7.2.2    TCP listening port .....	17
57	7.3    DNS server representation .....	17
58	7.3.1    Relationship with DNS service.....	17
59	7.3.2    UDP port for DNS resolution session .....	17
60	7.3.3    TCP port for DNS resolution session.....	17
61	7.4    DNS service configuration .....	18
62	7.4.1    DNS service endpoint configuration .....	18
63	7.5    DNS service relationship with IP interfaces.....	18
64	7.5.1    Modeling the IP interface over which a DNS resolution session is established .....	18
65	7.5.2    Modeling the IP interfaces for the DNS service.....	19
66	8    Methods.....	19
67	8.1    Extrinsic Methods.....	19
68	8.1.1    Job parameter.....	20
69	8.2    CIM_ProtocolService.ListenOnPort( ).....	20
70	8.3    CIM_ProtocolService.ListenOnUDPPort( ) .....	21
71	8.4    CIM_ProtocolService.RequestStateChange( ) .....	23
72	8.4.1    CIM_ProtocolService.RequestStateChange( ) conditional support.....	24
73	8.5    Profile conventions for operations .....	24
74	8.6    CIM_BindsTo .....	24
75	8.7    CIM_ElementCapabilities .....	24
76	8.8    CIM_ElementSettingData .....	25
77	8.8.1    CIM_ElementSettingData — ModifyInstance .....	25
78	8.9    CIM_HostedAccessPoint .....	25
79	8.10    CIM_HostedService .....	26
80	8.11    CIM_ProtocolService .....	26
81	8.11.1    CIM_ProtocolService — ModifyInstance .....	26
82	8.12    CIM_ProvidesEndpoint .....	26
83	8.13    CIM_ServiceAccessBySAP .....	27
84	8.14    CIM_IPProtocolEndpoint .....	27
85	8.15    CIM_TCPProtocolEndpoint.....	27
86	8.15.1    ModifyInstance.....	27

87	8.15.2	DeleteInstance .....	28
88	8.16	CIM_DNSServiceProtocolEndpoint .....	28
89	8.16.1	ModifyInstance .....	28
90	8.16.2	DeleteInstance .....	28
91	8.17	CIM_UDPProtocolEndpoint .....	28
92	8.17.1	ModifyInstance .....	29
93	8.17.2	DeleteInstance .....	29
94	8.18	CIM_DNSServiceCapabilities .....	29
95	8.19	CIM_DNSServiceSettingData .....	29
96	8.19.1	CIM_DNSServiceSettingData — ModifyInstance .....	29
97	9	Use cases .....	30
98	9.1	Profile Registration .....	30
99	9.2	Adding a UDP port for the DNS service .....	32
100	9.3	Determining the DNS service capabilities .....	33
101	9.3.1	Determine which IP address versions are supported .....	34
102	9.4	Configuring DNS service default settings .....	34
103	9.5	Modifying DNS service active settings .....	34
104	9.5.1	Enabling DNS service security policy .....	34
105	9.6	Viewing DNS service active settings .....	34
106	9.6.1	View the DNS service IP address .....	35
107	9.6.2	View the DNS resolution session policy .....	35
108	9.7	Stopping the DNS service on a specific UDP port .....	35
109	9.8	Disabling the DNS service .....	35
110	10	CIM Elements .....	35
111	10.1	CIM_BindsTo — CIM_IPProtocolEndpoint and CIM_TCPProtocolEndpoint .....	36
112	10.2	CIM_BindsTo — CIM_IPProtocolEndpoint and CIM_UDPProtocolEndpoint .....	37
113	10.3	CIM_BindsTo — CIM_TCPProtocolEndpoint and CIM_DNSServiceProtocolEndpoint .....	37
114	10.4	CIM_DNSServiceCapabilities .....	37
115	10.5	CIM_DNSServiceSettingData .....	38
116	10.6	CIM_DNSServiceProtocolEndpoint .....	39
117	10.7	CIM_ProtocolService .....	41
118	10.8	CIM_DNSServiceProtocolEndpointStats .....	41
119	10.9	CIM_ElementSettingData — CIM_ProtocolService and CIM_DNSServiceSettingData .....	42
120	10.10	CIM_ElementSettingData — CIM_DNSServiceProtocolEndpoint and CIM_DNSServiceSettingData .....	42
121	10.11	CIM_HostedAccessPoint — CIM_System and CIM_IPProtocolEndpoint .....	43
122	10.12	CIM_HostedAccessPoint — CIM_System and CIM_DNSServiceProtocolEndpoint .....	43
123	10.13	CIM_HostedService .....	44
124	10.14	CIM_IPProtocolEndpoint .....	44
125	10.15	CIM_RegisteredProfile .....	45
126	10.16	CIM_ElementCapabilities .....	45
127	10.17	CIM_ProvidesEndpoint .....	45
128	10.18	CIM_ServiceAccessBySAP - CIM_TCPProtocolEndpoint and CIM_ProtocolService .....	46
129	10.19	CIM_ServiceAccessBySAP - UDPProtocolEndpoint and CIM_ProtocolService .....	46
130	10.20	CIM_ElementConformsToProfile .....	47
131	10.21	CIM_TCPProtocolEndpoint .....	47
132	10.22	CIM_UDPProtocolEndpoint .....	47
133		ANNEX A (informative) Change log .....	49
134			
135			

136 **Figures**

137 Figure 1 – DNS Service Management Profile: Class diagram ..... 14

138 Figure 2 – Registered profile (ComputerSystem) ..... 31

139 Figure 3 - Registered profile (Network Management)..... 32

140 Figure 4 – UDP port configuration to specific interface ..... 33

141 **Tables**

142 Table 1 – Referenced profiles ..... 13

143 Table 2 – CIM\_ProtocolService.ListenOnPort( ) method: Return code values ..... 20

144 Table 3 – CIM\_ProtocolService.ListenOnPort( ) method: Parameters ..... 20

145 Table 4 – CIM\_ProtocolService. ListenOnUDPPort ( ) method: Return code values..... 22

146 Table 5 – CIM\_ProtocolService. ListenOnUDPPort ( ) method: Parameters ..... 22

147 Table 6 – CIM\_ProtocolService.RequestStateChange( ) method: Return code values ..... 23

148 Table 7 – CIM\_ProtocolService.RequestStateChange( ) method: Parameters ..... 23

149 Table 8 – Operations: CIM\_BindsTo..... 24

150 Table 9 – Operations: CIM\_ElementCapabilities ..... 24

151 Table 10 – Operations: CIM\_ElementSettingData..... 25

152 Table 11 – Operations: CIM\_HostedAccessPoint..... 25

153 Table 12 – Operations: CIM\_HostedService ..... 26

154 Table 13 – Operations: CIM\_ProtocolService..... 26

155 Table 14 – Operations: CIM\_ProvidesEndpoint..... 27

156 Table 15 – Operations: CIM\_ServiceAccessBySAP ..... 27

157 Table 16 – Operations: CIM\_TCPProtocolEndpoint ..... 27

158 Table 17 – Operations: CIM\_DNSServiceProtocolEndpoint..... 28

159 Table 18 – Operations: CIM\_UDPProtocolEndpoint..... 29

160 Table 19 – Operations: CIM\_DNSServiceSettingData ..... 29

161 Table 20 – CIM Elements: DNS service management profile ..... 35

162 Table 21 – Class: CIM\_BindsTo — CIM\_IPProtocolEndpoint and CIM\_TCPProtocolEndpoint ..... 36

163 Table 22 – Class: CIM\_BindsTo – CIM\_IPProtocolEndpoint and CIM\_UDPProtocolEndpoint..... 37

164 Table 23 – Class: CIM\_BindsTo – CIM\_TCPProtocolEndpoint and CIM\_DNSServiceProtocolEndpoint.. 37

165 Table 25 – Class: CIM\_DNSServiceCapabilities ..... 37

166 Table 26 – Class: CIM\_DNSServiceSettingData ..... 38

167 Table 27 – Class: CIM\_DNSServiceProtocolEndpoint ..... 40

168 Table 28 – Class: CIM\_ProtocolService ..... 41

169 Table 29 – Class: CIM\_DNSServiceProtocolEndpointStats ..... 42

170 Table 30 – Class: CIM\_ElementSettingData — CIM\_ProtocolService and CIM\_DNSServiceSettingData 42

171 Table 31 – Class: CIM\_ElementSettingData — CIM\_DNSServiceProtocolEndpoint and

172 CIM\_DNSServiceSettingData ..... 42

173 Table 32 – Class: CIM\_HostedAccessPoint — CIM\_System and CIM\_IPProtocolEndpoint ..... 43

174 Table 33 – Class: CIM\_HostedAccessPoint — CIM\_System and CIM\_DNSServiceProtocolEndpoint..... 44

175 Table 34 – Class: CIM\_HostedService ..... 44

176 Table 35 – Class: CIM\_IPProtocolEndpoint..... 44

177 Table 36 – Class: CIM\_RegisteredProfile ..... 45

178 Table 37 – Class: CIM\_ElementCapabilities ..... 45

179 Table 38 – Class: CIM\_ProvidesEndpoint ..... 46  
180 Table 39 – Class: CIM\_ServiceAccessBySAP - CIM\_TCPProtocolEndpoint and CIM\_ProtocolService... 46  
181 Table 40 – Class: CIM\_ServiceAccessBySAP - UDPProtocolEndpoint and CIM\_ProtocolService ..... 46  
182 Table 41 – Class: CIM\_ElementConformsToProfile ..... 47  
183 Table 42 – Class: CIM\_TCPProtocolEndpoint..... 47  
184 Table 43 – Class: CIM\_UDPProtocolEndpoint ..... 47  
185

186

## Foreword

187 The *DNS Service Management Profile* (DSP1069) was prepared by the Network Services Management  
188 Working Group of the DMTF.

189 DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems  
190 management and interoperability.

## 191 **Acknowledgments**

192 The DMTF acknowledges the following individuals for their contributions to this document:

193 Editors:

- 194 • ZhongYu Gu – ZTE Corporation
- 195 • Ghazanfar Ali – ZTE Corporation

196 Contributors:

- 197 • Hemal Shah – Broadcom Corporation
- 198 • John Crandall – Brocade Communications System
- 199 • Alex Zhdankin – Cisco Systems
- 200 • Steve Neely – Cisco Systems
- 201 • Shishir Pardikar – Citrix
- 202 • Eric Wells – Hitachi
- 203 • John Leung - Intel
- 204 • John Parchem – Microsoft Corporation
- 205 • Lawrence Lamers – VMware
- 206 • Dr. Bhumip Khasnabish - ZTE Corporation

207

## Introduction

208 The information in this specification should be sufficient for a provider or consumer of this data to identify  
209 unambiguously the classes, properties, methods, and values that shall be instantiated and manipulated to  
210 represent and manage Network Services and the associated configuration information. The target  
211 audience for this specification is implementers who are writing CIM-based providers or consumers of  
212 management interfaces that represent the component described in this document.

### 213 **Document conventions**

#### 214 **Typographical conventions**

#### 215 **Typographical conventions**

216 The following typographical conventions are used in this document:

- 217 • Document titles are marked in *italics*.
- 218 • ABNF rules are in `monospaced font`.



219

# DNS Service Management Profile

## 220 1 Scope

221 The *DNS Service Management Profile* is a profile that specifies the common information model (CIM)  
222 schema and use cases associated with the general and common aspects of managing the DNS server.  
223 This profile includes a specification of the configuration, protocol service, and protocol endpoint for the  
224 DNS server.

## 225 2 Normative references

226 The following referenced documents are indispensable for the application of this document. For dated or  
227 versioned references, only the edition cited (including any corrigenda or DMTF update versions) applies.  
228 For references without a date or version, the latest published edition of the referenced document  
229 (including any corrigenda or DMTF update versions) applies.

230 DMTF DSP0004, *CIM Infrastructure Specification 2.7*,  
231 [http://www.dmtf.org/standards/published\\_documents/DSP0004\\_2.7.pdf](http://www.dmtf.org/standards/published_documents/DSP0004_2.7.pdf)

232 DMTF DSP0200, *CIM Operations over HTTP 1.3*,  
233 [http://www.dmtf.org/standards/published\\_documents/DSP0200\\_1.3.pdf](http://www.dmtf.org/standards/published_documents/DSP0200_1.3.pdf)

234 DMTF DSP0223, *Generic Operations 1.0*,  
235 [http://www.dmtf.org/standards/published\\_documents/DSP0223\\_1.0.pdf](http://www.dmtf.org/standards/published_documents/DSP0223_1.0.pdf)

236 DMTF DSP1001, *Management Profile Specification Usage Guide 1.1*,  
237 [http://www.dmtf.org/standards/published\\_documents/DSP1001\\_1.1.pdf](http://www.dmtf.org/standards/published_documents/DSP1001_1.1.pdf)

238 DMTF DSP1033, *Profile Registration Profile 1.0*,  
239 [http://www.dmtf.org/standards/published\\_documents/DSP1033\\_1.0.pdf](http://www.dmtf.org/standards/published_documents/DSP1033_1.0.pdf)

240 DMTF DSP1053, *Base Metrics Profile 1.1*,  
241 [http://www.dmtf.org/standards/published\\_documents/DSP1053\\_1.1.pdf](http://www.dmtf.org/standards/published_documents/DSP1053_1.1.pdf)

242 DMTF DSP1036 *IP Interface Profile 1.1.1*,  
243 [http://www.dmtf.org/sites/default/files/standards/documents/DSP1036\\_1.1.1.pdf](http://www.dmtf.org/sites/default/files/standards/documents/DSP1036_1.1.1.pdf)

244 DMTF DSP1038 *DNS Client Profile 1.0.3*,  
245 [http://www.dmtf.org/sites/default/files/standards/documents/DSP1038\\_1.0.3.pdf](http://www.dmtf.org/sites/default/files/standards/documents/DSP1038_1.0.3.pdf)

246 DMTF DSP1097, *Virtual Ethernet Switch Profile 1.1*,  
247 [http://www.dmtf.org/sites/default/files/standards/documents/DSP1097\\_1.1.0.pdf](http://www.dmtf.org/sites/default/files/standards/documents/DSP1097_1.1.0.pdf)

248 DMTF DSP8020, *Message Registry XML Schema Specification 1.0*,  
249 [http://www.dmtf.org/standards/published\\_documents/DSP8020\\_1.0.xsd](http://www.dmtf.org/standards/published_documents/DSP8020_1.0.xsd)

250 IETF RFC 952, *DOD Internet Host Table Specification*, October 1985,  
251 <http://tools.ietf.org/html/rfc952>

252 IETF RFC 1034, *Domain Names – Concept and Facilities*, November 1987,  
253 <http://tools.ietf.org/html/rfc1034>

- 254 IETF RFC 1035, *Domain Names – Implementation and Specification*, November 1987,  
255 <http://tools.ietf.org/html/rfc1035>
- 256 IETF RFC 1208, *A Glossary of Networking Terms*, March 1991,  
257 <http://tools.ietf.org/html/rfc1208>
- 258 IETF RFC 1611, *DNS Server MIB Extensions*, May 1994,  
259 <http://tools.ietf.org/html/rfc1611>
- 260 IETF RFC 1996, *A Mechanism for Prompt Notification of Zone Changes (DNS NOTIFY)*, August 1996,  
261 <http://tools.ietf.org/html/rfc1996>
- 262 IETF RFC 2671, *Extension Mechanisms for DNS (EDNS0)*, August 1999,  
263 <http://www.ietf.org/rfc/rfc2671.txt>
- 264 IETF RFC 2845, *Secret Key Transaction Authentication for DNS (TSIG)*, May 2000,  
265 <https://tools.ietf.org/html/rfc2845>
- 266 IETF RFC 3007, *Dynamic Updates in the Domain Name System (DNS UPDATE)*, November 2000,  
267 <http://tools.ietf.org/html/rfc3007>
- 268 IETF RFC 3596, *DNS Extensions to Support IP Version 6*, October 2003,  
269 <https://tools.ietf.org/html/rfc3596>
- 270 IETF RFC4033, *DNS Security Introduction and Requirements*, March 2005,  
271 <https://tools.ietf.org/html/rfc4033>
- 272 IETF RFC4034, *Resource Records for the DNS Security Extensions*, March 2005,  
273 <https://tools.ietf.org/html/rfc4034>
- 274 IETF RFC4035, *Protocol Modifications for the DNS Security Extensions*, March 2005,  
275 <https://tools.ietf.org/html/rfc4035>
- 276 IETF RFC 4291, *IP Version 6 Addressing Architecture*, February 2006,  
277 <http://www.ietf.org/rfc/rfc4291.txt>
- 278 ISO/IEC Directives, Part 2, *Rules for the structure and drafting of International Standards*,  
279 <http://isotc.iso.org/livelink/livelink.exe?func=ll&objId=4230456&objAction=browse&sort=subtype>

### 280 **3 Terms and definitions**

281 In this document, some terms have a specific meaning beyond the normal English meaning. Those terms  
282 are defined in this clause.

283 The terms "shall" ("required"), "shall not", "should" ("recommended"), "should not" ("not recommended"),  
284 "may", "need not" ("not required"), "can" and "cannot" in this document are to be interpreted as described  
285 in [ISO/IEC Directives, Part 2](#), Annex H. The terms in parentheses are alternatives for the preceding term,  
286 for use in exceptional cases when the preceding term cannot be used for linguistic reasons. Note that  
287 [ISO/IEC Directives, Part 2](#), Annex H specifies additional alternatives. Occurrences of such additional  
288 alternatives shall be interpreted in their normal English meaning.

289 The terms "clause", "subclause", "paragraph", and "annex" in this document are to be interpreted as  
290 described in [ISO/IEC Directives, Part 2](#), Clause 5.

291 The terms "normative" and "informative" in this document are to be interpreted as described in [ISO/IEC](#)  
292 [Directives, Part 2](#), Clause 3. In this document, clauses, subclauses, or annexes labeled "(informative)" do  
293 not contain normative content. Notes and examples are always informative elements.

294 The terms defined in [DSP0004](#), [DSP0223](#), and [DSP1001](#) apply to this document. The following additional  
295 terms are used in this document.

296 **3.1**  
297 **conditional**

298 indicates requirements to be followed strictly to conform to the document when the specified conditions  
299 are met

300 **3.2**  
301 **mandatory**

302 indicates requirements to be followed strictly to conform to the document and from which no deviation is  
303 permitted

304 **3.3**  
305 **optional**

306 indicates a course of action permissible within the limits of the document

307 **3.4**  
308 **pending configuration**

309 indicates the configuration that will be applied to an IP network connection the next time the IP network  
310 connection accepts a configuration

311 **3.5**  
312 **referencing profile**

313 indicates a profile that owns the definition of this class and can include a reference to this profile in its  
314 "Referenced Profiles" table

315 **3.6**  
316 **unspecified**

317 indicates that this profile does not define any constraints for the referenced CIM element or operation

318 **4 Symbols and abbreviated terms**

319 The abbreviations defined in [DSP0004](#), [DSP0223](#), and [DSP1001](#) apply to this document. The following  
320 additional abbreviations are used in this document.

321 **4.1**  
322 **IP**  
323 Internet Protocol

324 **4.2**  
325 **DNS**  
326 Domain Name System

327 **4.3**  
328 **EDNS0**  
329 Extension mechanisms for DNS

330 **4.4**

331 **DNSSEC**

332 The Domain Name System Security Extensions (DNSSEC) is a suite of Internet Engineering Task Force  
333 (IETF) specifications for securing certain kinds of information provided by the DNS as used on IP  
334 networks

335 **4.5**

336 **TCP**

337 Transmission Control Protocol

338 **4.6**

339 **TSIG**

340 Transaction SIGnature - is a computer networking protocol defined in RFC 2845

341 **4.7**

342 **UDP**

343 User Datagram Protocol

344 **5 Synopsis**

345 **Profile name:** DNS Service Management Profile

346 **Version:** 1.0.0

347 **Organization:** DMTF

348 **CIM Schema version:** 2.46

349 **Central class:** CIM\_ProtocolService

350 **Scoping class:** CIM\_System

351 The *DNS Service Management Profile* is a profile that specifies the CIM schema and use cases  
352 associated with managing of the DNS server. This profile includes a specification for configuration and life  
353 cycle management of the DNS service.

354 Table 1 identifies profiles on which this profile has a dependency.

355

**Table 1 – Referenced profiles**

Profile Name	Organization	Version	Requirement	Description
Profile Registration	DMTF	1.0	Mandatory	None
IP Configuration Profile	DMTF	1.0	Optional	None
IP Interface Profile	DMTF	1.1.1	Mandatory	None

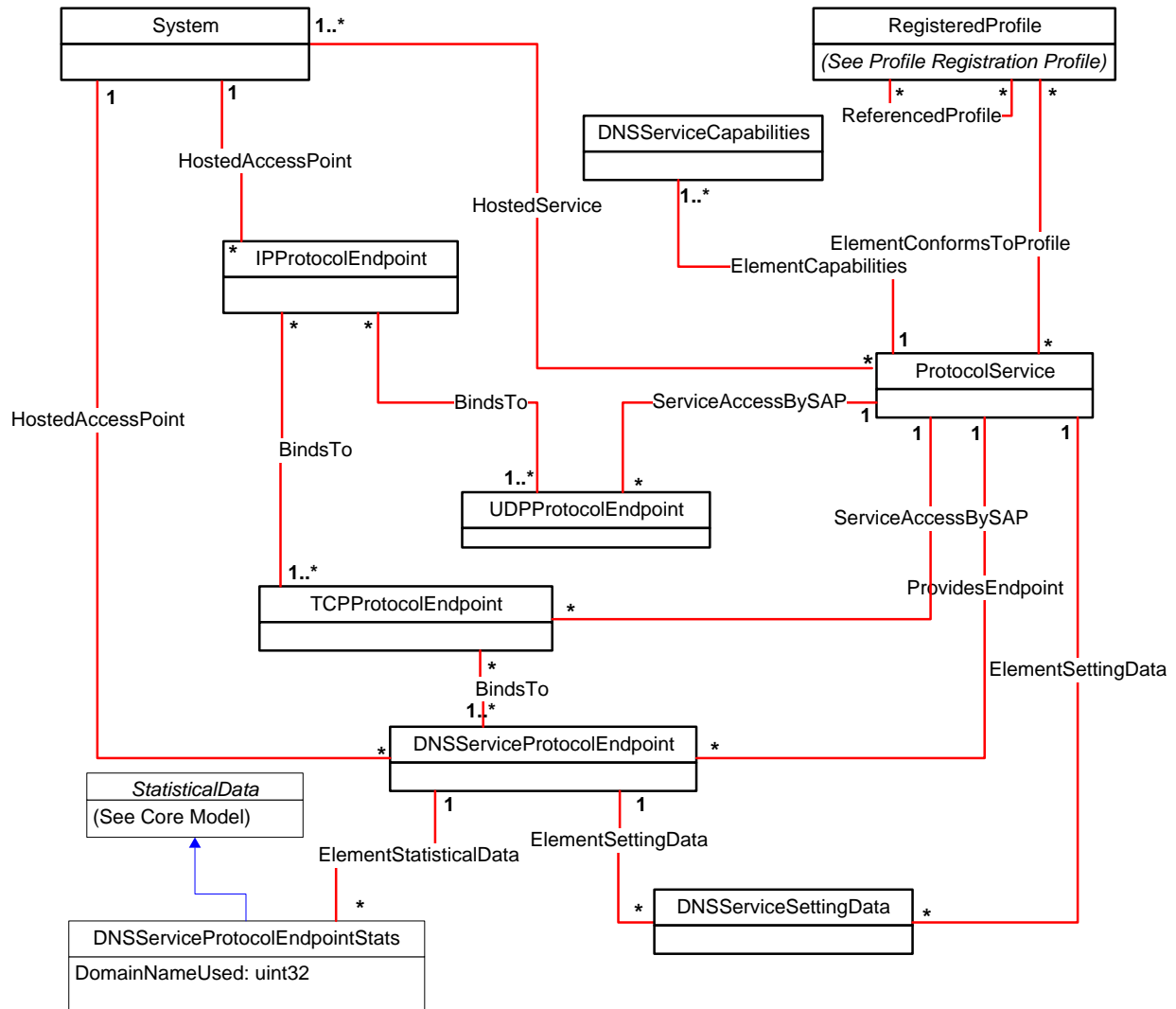
356 The central class for the *DNS Service Management Profile* shall be the CIM\_ProtocolService class. The  
 357 central instance of the *DNS Service Management Profile* shall be an instance of CIM\_ProtocolService.  
 358 The scoping class for the *DNS Service Management Profile* shall be CIM\_System. The scoping instance  
 359 of the *DNS Service Management Profile* shall be the instance of CIM\_System to which the central  
 360 instance is associated through an instance of the CIM\_HostedService association.

361 **6 Description**

362 The *DNS Service Management Profile* is a profile that specifies the CIM schema and use cases  
 363 associated with the general and common aspects of managing DNS service. This profile includes a  
 364 specification of the configuration, protocol service, and protocol endpoint for the DNS service.

365 **6.1 Class diagram**

366 Figure 1 represents the class schema for the *DNS Service Management Profile*. For simplicity, the CIM\_  
 367 prefix has been removed from the names of the classes.



368  
369

370 **Figure 1 - DNS Service Management Profile: Class diagram**

371 Functionality within the scope of this profile includes:

- 372 • representation of the DNS service
- 373 • configuration of the DNS service
- 374 • representation of protocol endpoints of the DNS service
- 375 • remote representation of DNS clients

376 Functionality explicitly excluded from the scope of this profile includes modeling of the DNS client. The  
377 DNS clients are modeled in [DSP1038](#).

378 The DNS service is represented by an instance of CIM\_ProtocolService. The capabilities of the DNS  
379 service are represented by an instance of CIM\_DNSServiceCapabilities. The current configuration of the  
380 DNS service is modeled with the properties from the instance of CIM\_DNSServiceSettingData. The  
381 management access to the DNS service is represented by CIM\_DNSServiceProtocolEndpoint. DNS

382 service uses UDP (CIM\_UDPProtocolEndpoint) and optionally TCP (CIM\_TCPProtocolEndpoint) for DNS  
383 query/response and requires TCP for certain DNS inter-service activities (e.g., zone transfer) [RFC1035].

### 384 6.1.1 Support for IP protocol versioning

385 The DNS service integrates mechanisms to support both IPv4 and IPv6 (simultaneously) as described in  
386 RFC3596. The IP protocol version used for querying resource records is independent of the protocol  
387 version of the resource records i.e., IPv4 transport can be used to query IPv6 records and vice versa.

### 388 6.1.2 Management of DNS security

389 TSIG (RFC2845) is used to securely authenticate transactions and is configurable.

390 The DNS Security Extensions (RFC4033, RFC4034, and RFC4035) are used to protect the integrity of  
391 data in the DNS by establishing a chain of trust. DNS data (e.g., zone) is digitally signed to attest its  
392 validity.

393 TSIG and DNSSEC can be used independently. DNS security management aspects are outside of the  
394 scope of this specification.

### 395 6.1.3 Representation of DNS service usage data

396 The CIM\_DNSServiceProtocolEndpointStats represents statistics of operations of the DNS service.

## 397 7 Implementation

398 NOTE: Section 7 is normative and should be written from the class description viewpoint, not methods.  
399 Also some of the current content of this section (content below) should be moved to non-normative  
400 section 6. [Ghazanfar: I modified text by making normative statements. Class description is given in  
401 clause 10. Further review is needed to find relevant text to move (if any)]. [Ghazanfar: Done]

### 402 7.1 Representing a DNS service

403 An instance of CIM\_ProtocolService shall represent the DNS service being modeled.

#### 404 7.1.1 CIM\_ProtocolService.Protocol

405 The Protocol property of the CIM\_ProtocolService instance shall have a value of X (DNS).

406 NOTE: Do we have a value for X already defined? Is it in the schema (CR?)

407 [Ghazanfar: There is NSMWG Mantis to address this issue.]

#### 408 7.1.2 DNS service capabilities

409 An instance of CIM\_DNSServiceCapabilities shall be associated with the CIM\_ProtocolService instance  
410 through an instance of CIM\_ElementCapabilities. This instance of CIM\_DNSServiceCapabilities shall  
411 represent the capabilities of the DNS service.

#### 412 7.1.3 Managing the DNS service's state

413 This clause describes the usage of the RequestedState and EnabledState properties to represent the  
414 state of an instance of CIM\_ProtocolService.

**415 7.1.3.1 State management supported**

416 Exactly one instance of CIM\_DNSServiceCapabilities shall be associated with an instance of  
417 CIM\_ProtocolService, which indicates support for managing the state of the DNS service.

418 Support for managing the state of the DNS service is conditional behavior. This clause describes the CIM  
419 elements and behaviors that shall be implemented when this behavior is supported.

**420 7.1.3.2 RequestedState**

421 The CIM\_ProtocolService.RequestedState property shall have one of the values specified in the  
422 CIM\_DNSServiceCapabilities.RequestedStatesSupported property or a value of 5 (No Change).

423 When the RequestedStatesProperty of the associated instance of  
424 CIM\_EnabledLogicalElementCapabilities does not contain any values, the RequestedState property shall  
425 have the value of 12 (Not Applicable).

**426 7.1.3.3 EnabledState**

427 The EnabledState property shall have the value 2 (Enabled), 3 (Disabled), or 6 (Enabled but Offline).  
428 Enabled (2) indicates that the element is or could be executing commands, will process any queued  
429 commands, and queues new requests. Disabled (3) indicates that the element will not execute  
430 commands and will drop any new requests. Enabled but Offline (6) indicates that the element might be  
431 completing commands, and will drop any new requests.

**432 7.1.3.4 Indicating state management support with CIM\_DNSServiceCapabilities**

433 When state management is supported, the RequestedStatesSupported property of the  
434 CIM\_DNSServiceCapabilities instance associated with the CIM\_ProtocolService instance via an instance  
435 of CIM\_ElementCapabilities shall contain at least one value. The RequestedStatesSupported property  
436 may have zero or more of the following values: 2 (Enabled), 3 (Disabled), or 11 (Reset).

**437 7.2 DNS service port management****438 7.2.1 UDP port**

439 An implementation may model one or more UDP ports of the DNS service. When the implementation  
440 models the UDP ports, the following requirements shall apply for each UDP port.

**441 7.2.1.1 CIM\_UDPProtocolEndpoint**

442 There shall be an instance of CIM\_UDPProtocolEndpoint in which the PortNumber property of the  
443 instance indicates the UDP port number on which the DNS service is accessible.

**444 7.2.1.2 Relationship of UDP port to the DNS service**

445 An instance of CIM\_ServiceAccessBySAP shall associate the CIM\_ProtocolService instance with the  
446 CIM\_UDPProtocolEndpoint instance.

**447 7.2.1.3 Managing UDP ports**

448 The implementation may support managing the UDP ports on which the DNS service is accessible. The  
449 ListenOnUDPPort() method of the CIM\_ProtocolService class shall be used to add UDP ports on which  
450 the DNS service will be accessible. Using the DeleteInstance intrinsic operation to remove an instance of  
451 CIM\_UDPProtocolEndpoint shall remove the DNS service from being accessible on the represented port.



## 452 7.2.2 TCP listening port

453 An implementation may model one or more listening ports of the DNS service. When the implementation  
454 models the listening ports, the following requirements shall apply to each listening port.

### 455 7.2.2.1 CIM\_TCPProtocolEndpoint

456 There shall be an instance of CIM\_TCPProtocolEndpoint in which the PortNumber property of the  
457 instance shall indicate the TCP port number on which the DNS service is listening.

### 458 7.2.2.2 Relationship of TCP port to the DNS service

459 An instance of CIM\_ServiceAccessBySAP shall associate the CIM\_ProtocolService instance with the  
460 CIM\_TCPProtocolEndpoint instance.

### 461 7.2.2.3 Managing TCP listening ports

462 The implementation may support managing the ports on which the DNS service listens. The  
463 ListenOnPort( ) method of the CIM\_ProtocolService class shall be used to add ports on which the DNS  
464 service will listen. Using the DeleteInstance intrinsic operation to delete an instance of  
465 CIM\_TCPProtocolEndpoint will stop the DNS service from listening on the represented port.

## 466 7.3 DNS server representation

467 The management access to the DNS service shall be modeled using an instance of  
468 CIM\_DNSServiceProtocolEndpoint.

### 469 7.3.1 Relationship with DNS service

470 An instance of CIM\_ProvidesEndpoint shall associate the CIM\_ProtocolService with the  
471 CIM\_DNSServiceProtocolEndpoint.

### 472 7.3.2 UDP port for DNS resolution session

473 An implementation may model the UDP port to which the DNS resolution session is bound. When the  
474 implementation models the UDP port, the following requirements shall apply.

#### 475 7.3.2.1 CIM\_UDPProtocolEndpoint

476 When the UDP port on which the DNS resolution session is bound is modeled, the UDP port shall be  
477 modeled using an instance of CIM\_UDPProtocolEndpoint.

### 478 7.3.3 TCP port for DNS resolution session

479 An implementation may model the TCP port on which the DNS maintenance is bound. When the  
480 implementation models the TCP port, the following requirements shall apply.

#### 481 7.3.3.1 CIM\_TCPProtocolEndpoint

482 When the TCP port to which the DNS resolution session is bound is modeled, the TCP port shall be  
483 modeled using an instance of CIM\_TCPProtocolEndpoint.

## 484 7.4 DNS service configuration

485 The default configuration is the configuration of the DNS service when it was first installed on the  
486 managed system. When an implementation exposes the default configuration, the default configuration  
487 shall be represented by an instance of CIM\_DNSServiceSettingData that is associated with the  
488 CIM\_ProtocolService through an instance of CIM\_ElementSettingData, where the IsDefault property of  
489 the CIM\_ElementSettingData instance has a value of 1 (Is Default).

### 490 7.4.1 DNS service endpoint configuration

491 When a DNS resolution session is created, it will have an initial configuration. Implementations can  
492 indicate to DNS clients the configuration that will be assigned to a DNS resolution session.  
493 Implementations can also indicate to DNS clients the current configuration that an active DNS resolution  
494 session had when the DNS resolution session was first established.

#### 495 7.4.1.1 Initial configuration that will be assigned

496 An implementation may assign the same initial configuration for all DNS resolution sessions that are  
497 spawned. When the implementation assigns the same initial configuration for all DNS resolution sessions,  
498 the configuration that a DNS resolution session will have when it is established shall be represented by  
499 an instance of CIM\_DNSServiceSettingData that is associated with the instance of CIM\_ProtocolService  
500 through an instance of CIM\_ElementSettingData, where the IsNext property of the  
501 CIM\_ElementSettingData instance has a value of 1 (Is Next).

#### 502 7.4.1.2 Current configuration of a DNS resolution session

503 The initial configuration of a DNS resolution session may be modeled. When the configuration that a DNS  
504 resolution session had when it was established is modeled, it shall be represented by an instance of  
505 CIM\_DNSServiceSettingData that is associated with the instance of CIM\_DNSServiceProtocolEndpoint  
506 through an instance of CIM\_ElementSettingData, where the IsCurrent property of the  
507 CIM\_ElementSettingData instance has a value of 1 (Is Current).

508 A discrete instance of CIM\_DNSServiceSettingData is not required for each active DNS resolution  
509 session. The instance of CIM\_DNSServiceSettingData that is associated with the instance of  
510 CIM\_DNSServiceProtocolEndpoint needs only to accurately reflect the initial configuration of the DNS  
511 resolution session.

## 512 7.5 DNS service relationship with IP interfaces

513 This clause details requirements for specifying the relationship between the DNS service or DNS  
514 resolution session and one or more IP interfaces of the system.

### 515 7.5.1 Modeling the IP interface over which a DNS resolution session is established

516 When the specific port for a DNS resolution is modeled, the specific IP interface over which the DNS  
517 resolution is active shall be modeled.

518 When the implementation models the specific interface over which a DNS resolution is active on TCP  
519 port, there shall be an instance of the CIM\_BindsTo association where the value of the Antecedent  
520 property shall be a reference to the CIM\_IPProtocolEndpoint instance and the value of the Dependent  
521 property shall be a reference to the CIM\_TCPProtocolEndpoint instance.

522 When the implementation models the specific interface over which a DNS resolution is active on UDP  
523 port, there shall be an instance of the CIM\_BindsTo association where the value of the Antecedent

524 property shall be a reference to the CIM\_IPProtocolEndpoint instance and the value of the Dependent  
525 property shall be a reference to the CIM\_UDPProtocolEndpoint instance.

## 526 7.5.2 Modeling the IP interfaces for the DNS service

527 When the specific port for a DNS service is modeled, the specific IP interface over which the service is  
528 accessible may be modeled. This behavior is optional.

529 When the implementation models the specific interface over which a DNS service is active on TCP port,  
530 there shall be an instance of the CIM\_BindsTo association where the value of the Antecedent property  
531 shall be a reference to the CIM\_IPProtocolEndpoint instance and the value of the Dependent property  
532 shall be a reference to the CIM\_TCPProtocolEndpoint instance.

533 When the implementation models the specific interface over which a DNS service is accessible on UDP  
534 port, there shall be an instance of the CIM\_BindsTo association where the value of the Antecedent  
535 property shall be a reference to the CIM\_IPProtocolEndpoint instance and the value of the Dependent  
536 property shall be a reference to the CIM\_UDPProtocolEndpoint instance.

537 When the CIM\_TCPProtocolEndpoint or CIM\_UDPProtocolEndpoint is not associated with one or more  
538 instances of CIM\_IPProtocolEndpoint through an instance of CIM\_BindsTo, the DNS service accepts  
539 connections over all the IP interfaces of the system.

## 540 8 Methods

541 This clause details the requirements for supporting intrinsic operations and extrinsic methods for the CIM  
542 elements defined by this profile.

### 543 8.1 Extrinsic Methods

544 If synchronous execution of a method succeeds, the implementation shall set a return value of  
545 0 (Completed with No Error).

546 If synchronous execution of a method fails, the implementation shall set a return value of 2 (Failed) or a  
547 more specific return code as specified with the respective method.

548 If a method is executed as an asynchronous task, the implementation shall perform all of the following ac-  
549 tions:

- 550 • Set a return value of 4096 (Job Started).
- 551 • Set the value of the Job output parameter to refer to an instance of the CIM\_ConcreteJob class  
552 that represents the asynchronous task.
- 553 • Set the values of the JobState and TimeOfLastStateChange properties in that instance to repre-  
554 sent the state and last state change time of the asynchronous task.

555 In addition, the implementation may present state change indications as task state changes occur.

556 If the method execution as an asynchronous task succeeds, the implementation shall perform all of the  
557 following actions:

- 558 • Set the value of the JobState property to 7 (Completed).
- 559 • Provide an instance of the CIM\_AffectedJobEntity association with property values set as fol-  
560 lows: **NOTE – we need to use example in the highlighted section relevant to THIS spec.**  
561 **[Ghazanfar: Please see below text for consideration]**

- 562           – The value of the AffectedElement property shall refer to the object that represents the top-  
563 level entity that was created or modified by the asynchronous task. For example, for the  
564 CIM\_ProtocolService.AddDNSServiceProtocolEndpoint() method, this is an instance of  
565 the CIM\_DNSProtocolServiceEndpoint class
- 566           – The value of the AffectingElement property shall refer to the instance of the  
567 CIM\_ConcreteJob class that represents the completed asynchronous task.
- 568           – The value of the first element in the ElementEffects[ ] array property (ElementEffects[0])  
569 shall be set to 5 (Create) for the CIM\_ProtocolService.AddDNSServiceProtocolEndpoint()  
570 method. Otherwise, this value shall be 0 (Unknown).

571 If the method execution as an asynchronous task fails, the implementation shall set the value of the  
572 JobState property to 9 (Killed) or 10 (Exception).

### 573 8.1.1 Job parameter

574 The implementation shall set the value of the Job parameter as a result of an asynchronous execution of  
575 a method of the CIM\_ProtocolService as follows:

- 576           • If the method execution is performed synchronously, the implementation shall set the value to  
577 NULL.
- 578           • If the method execution is performed asynchronously, the implementation shall set the value to  
579 refer to the instance of the CIM\_ConcreteJob class that represents the asynchronous task.

### 580 8.2 CIM\_ProtocolService.ListenOnPort( )

581 The CIM\_ProtocolService.ListenOnPort( ) method shall be supported when the  
582 ListeningPortManagement property of the associated instance of CIM\_DNSServiceCapabilities has a  
583 value of TRUE. When the value of the ListeningPortManagement property is FALSE, the  
584 CIM\_ProtocolService.ListenOnPort( ) method shall not be supported.

585 The CIM\_ProtocolService.ListenOnPort( ) method is used to configure TCP ports on which the  
586 CIM\_ProtocolService instance will listen. Detailed requirements of the ListenOnPort() method are  
587 specified in Table 2 and Table 3.

588 No standard messages are defined.

589

590 **Table 2 – CIM\_ProtocolService.ListenOnPort( ) method: Return code values**

Value	Description
0	Request was successfully executed.
1	Method is unsupported in the implementation.
2	Error occurred.
0x1000	Job started: REF returned to started CIM_ConcreteJob

591 **Table 3 – CIM\_ProtocolService.ListenOnPort( ) method: Parameters**

Qualifiers	Name	Type	Description/Values
IN	IPEndpoint	CIM_IPProtocolEndpoint REF	Optional reference to the specific CIM_IPProtocolEndpoint instance to which the created

Qualifiers	Name	Type	Description/Values
			CIM_TCPProtocolEndpoint instance will be bound
OUT	TCPEndpoint	CIM_TCPProtocolEndpoint REF	Reference to the CIM_TCPProtocolEndpoint instance that is created if the method is successful
IN, REQ	PortNumber	uint16	Desired port number for the service to listen

592 When the method is completed successfully, the implementation shall create an instance of  
 593 CIM\_TCPProtocolEndpoint. The value of the PortNumber property of the instance of  
 594 CIM\_TCPProtocolEndpoint shall be the value of the PortNumber parameter of the method invocation.  
 595 The implementation shall create an instance of CIM\_ServiceAccessBySAP that references the instance  
 596 of CIM\_TCPProtocolEndpoint and references the instance of CIM\_ProtocolService on which the method  
 597 was invoked.

598 The implementation shall perform the following actions when the IPEndpoint parameter is not specified:

- 599 • The implementation shall create an instance of CIM\_HostedAccessPoint that references the  
 600 newly created CIM\_TCPProtocolEndpoint instance and the instance of CIM\_System with which  
 601 the CIM\_ProtocolService instance is associated through an instance of CIM\_HostedService (the  
 602 scoping system).
- 603 • For each instance of CIM\_IPProtocolEndpoint that is associated through an instance of  
 604 CIM\_HostedAccessPoint with the CIM\_System instance with which the instance of  
 605 CIM\_ProtocolService on which this method was invoked is associated through an instance of  
 606 CIM\_HostedService, the implementation shall create an instance of the CIM\_BindsTo  
 607 association where the value of the Antecedent property shall be a reference to the  
 608 CIM\_IPProtocolEndpoint instance and the value of the Dependent property shall be a reference  
 609 to the CIM\_TCPProtocolEndpoint instance.

610 The implementation shall perform the following actions when the IPEndpoint parameter is specified:

- 611 • The implementation shall create an instance of CIM\_HostedAccessPoint that references the  
 612 newly created CIM\_TCPProtocolEndpoint instance and the instance of CIM\_System with which  
 613 the CIM\_IPProtocolEndpoint instance is associated through an instance of  
 614 CIM\_HostedAccessPoint.
- 615 • The implementation shall create an instance of the CIM\_BindsTo association where the value of  
 616 the Antecedent property shall be a reference to the CIM\_IPProtocolEndpoint instance and the  
 617 value of the Dependent property shall be a reference to the CIM\_TCPProtocolEndpoint  
 618 instance.

619 **8.3 CIM\_ProtocolService.ListenOnUDPPort( )**

620 The CIM\_ProtocolService.ListenOnUDPPort( ) method shall be supported when the  
 621 ListeningPortManagement property of the associated instance of CIM\_DNSServiceCapabilities has a  
 622 value of TRUE. When the value of the ListeningPortManagement property is FALSE, the  
 623 CIM\_ProtocolService.ListenOnUDPPort( ) method shall not be supported.

624 The CIM\_ProtocolService.ListenOnUDPPort( ) method is used to configure UDP port on which the  
 625 CIM\_ProtocolService instance will access DNS service. Detailed requirements of the  
 626 ListenOnUDPPort( ) method are specified in Table 4 and Table 5.

627 No standard messages are defined.

628 **Table 4 – CIM\_ProtocolService. ListenOnUDPPort ( ) method: Return code values**

Value	Description
0	Request was successfully executed.
1	Method is unsupported in the implementation.
2	Error occurred.
0x1000	Job started: REF returned to started CIM_ConcreteJob

629 **Table 5 – CIM\_ProtocolService. ListenOnUDPPort ( ) method: Parameters**

Qualifiers	Name	Type	Description/Values
IN	IPEndpoint	CIM_IPProtocolEndpoint REF	Optional reference to the specific CIM_IPProtocolEndpoint instance to which the created CIM_UDPProtocolEndpoint instance will be bound
OUT	UDPEndpoint	CIM_UDPProtocolEndpoint REF	Reference to the CIM_UDPProtocolEndpoint instance that is created if the method is successful
IN, REQ	PortNumber	uint16	Desired port number for the service to access

630 When the method is completed successfully, the implementation shall create an instance of  
 631 CIM\_UDPProtocolEndpoint. The value of the PortNumber property of the instance of  
 632 CIM\_UDPProtocolEndpoint shall be the value of the PortNumber parameter of the method invocation.  
 633 The implementation shall create an instance of CIM\_ServiceAccessBySAP that references the instance  
 634 of CIM\_UDPProtocolEndpoint and references the instance of CIM\_ProtocolService on which the method  
 635 was invoked.

636 The implementation shall perform the following actions when the IPEndpoint parameter is not specified:

- 637 • The implementation shall create an instance of CIM\_HostedAccessPoint that references the  
 638 newly created CIM\_UDPProtocolEndpoint instance and the instance of CIM\_System with which  
 639 the CIM\_ProtocolService instance is associated through an instance of CIM\_HostedService (the  
 640 scoping system).
- 641 • For each instance of CIM\_IPProtocolEndpoint that is associated through an instance of  
 642 CIM\_HostedAccessPoint with the CIM\_System instance with which the instance of  
 643 CIM\_ProtocolService on which this method was invoked is associated through an instance of  
 644 CIM\_HostedService, the implementation shall create an instance of the CIM\_BindsTo  
 645 association where the value of the Antecedent property shall be a reference to the  
 646 CIM\_IPProtocolEndpoint instance and the value of the Dependent property shall be a reference  
 647 to the CIM\_UDPProtocolEndpoint instance.

648 The implementation shall perform the following actions when the IPEndpoint parameter is specified:

- 649 • The implementation shall create an instance of CIM\_HostedAccessPoint that references the  
 650 newly created CIM\_UDPProtocolEndpoint instance and the instance of CIM\_System with which  
 651 the CIM\_IPProtocolEndpoint instance is associated through an instance of  
 652 CIM\_HostedAccessPoint.
- 653 • The implementation shall create an instance of the CIM\_BindsTo association where the value of  
 654 the Antecedent property shall be a reference to the CIM\_IPProtocolEndpoint instance and the

655 value of the Dependent property shall be a reference to the CIM\_UDPProtocolEndpoint  
 656 instance.

657 **8.4 CIM\_ProtocolService.RequestStateChange( )**

658 Invocation of the CIM\_ProtocolService.RequestStateChange( ) method will change the element's state to  
 659 the value specified in the RequestedState parameter. The Enabled and Disabled values of the  
 660 RequestedState parameter correspond to enabling or disabling the functionality represented by the  
 661 instance of CIM\_ProtocolService. A value of 2 (Enabled) shall correspond to a request to enable the  
 662 functionality. A value of 3 (Disabled) shall correspond to a request to disable the functionality. A value of  
 663 11 (Reset) shall initiate a reset of the DNS service.

664 See 7.1.3.2 for information about the effect of this method on the RequestedState property.

665 The method shall be considered successful if the availability of the functionality upon completion of the  
 666 method corresponds to the desired availability indicated by the RequestedState parameter. An actual  
 667 change in state does not need to occur for the method to be considered successful. It is sufficient that the  
 668 resultant state be equal to the requested state. Upon successful completion of the method, the Return  
 669 Value shall be zero.

670 See 7.1.3.3 for information about the effect of this method on the EnabledState property.

671 Detailed requirements of the RequestStateChange( ) method are specified in Table 6 and Table 7.

672 No standard messages are defined.

673 Invoking the CIM\_ProtocolService.RequestStateChange( ) method multiple times could result in earlier  
 674 requests being overwritten or lost.

675 **Table 6 – CIM\_ProtocolService.RequestStateChange( ) method: Return code values**

Value	Description
0	Request was successfully executed.
1	Method is unsupported in the implementation.
2	Error occurred.
0x1000	Job started: REF returned to started CIM_ConcreteJob

676 **Table 7 – CIM\_ProtocolService.RequestStateChange( ) method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	RequestedState	uint16	Valid state values: 2 (Enabled) 3 (Disabled) 11 (Reset)
OUT	Job	CIM_ConcreteJob REF	Returned if job started
IN, REQ	TimeoutPeriod	datetime	Client specified maximum amount of time that the transition to a new state is supposed to take: 0 or NULL – No time requirements <interval> – Maximum time allowed

#### 677 8.4.1 CIM\_ProtocolService.RequestStateChange( ) conditional support

678 Support for the RequestStateChange( ) method is conditional on the indication of support for at least one  
 679 value for the RequestedState parameter as advertised through the RequestedStatesSupported property  
 680 of an associated instance of CIM\_EnabledLogicalElementCapabilities. When the  
 681 CIM\_EnabledLogicalElementCapabilities.RequestedStatesSupported property contains at least one  
 682 value, the CIM\_ProtocolService.RequestStateChange( ) method shall be implemented and supported.  
 683 The CIM\_ProtocolService.RequestStateChange( ) method shall not return a value of 1 (Unsupported).

### 684 8.5 Profile conventions for operations

685 For each profile class (including associations), the implementation requirements for operations, including  
 686 those in the following default list, are specified in class-specific subclauses of this clause.

687 The default list of operations is as follows:

- 688 • GetInstance
- 689 • EnumerateInstances
- 690 • EnumerateInstanceNames
- 691 • Associators
- 692 • AssociatorNames
- 693 • References
- 694 • ReferenceNames

### 695 8.6 CIM\_BindsTo

696 Table 8 lists operations that either have special requirements beyond those from [DSP0200](#) or shall not be  
 697 supported.

698 **Table 8 – Operations: CIM\_BindsTo**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

### 699 8.7 CIM\_ElementCapabilities

700 Table 9 lists implementation requirements for operations. If implemented, these operations shall be  
 701 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 9, all operations in  
 702 the default list in 8.3 shall be implemented as defined in [DSP0200](#).

703 **Table 9 – Operations: CIM\_ElementCapabilities**

Operation	Requirement	Messages
Associators	Unspecified	None



Operation	Requirement	Messages
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

704 **8.8 CIM\_ElementSettingData**

705 Table 10 lists operations that either have special requirements beyond those from [DSP0200](#) or shall not  
706 be supported.

707 **Table 10 – Operations: CIM\_ElementSettingData**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

708 **8.8.1 CIM\_ElementSettingData — ModifyInstance**

709 When an instance of CIM\_ElementSettingData associates an instance of CIM\_DNSServiceSettingData  
710 with an instance of CIM\_DNSServiceProtocolEndpoint, the following rules shall govern the behavior of  
711 the ModifyInstance operation:

- 712 • The ModifyInstance operation shall not allow the IsDefault property to be modified.
- 713 • The ModifyInstance operation shall not allow the IsCurrent property to be modified.
- 714 • When the ModifyInstance operation is used to modify the IsNext property to a value of 1 (Is  
715 Next), the ModifyInstance operation shall implement the following behavior:
  - 716 – The ModifyInstance operation shall find all other instances of CIM\_ElementSettingData  
717 that associate a CIM\_DNSServiceSettingData instance with the  
718 CIM\_DNSServiceProtocolEndpoint instance referenced by the target instance of  
719 CIM\_ElementSettingData.
  - 720 – For each instance of CIM\_ElementSettingData found, the ModifyInstance operation shall  
721 modify the value of its IsNext property to a value of 2 (Is Not Next).

722 **8.9 CIM\_HostedAccessPoint**

723 Table 11 lists operations that either have special requirements beyond those from [DSP0200](#) or shall not  
724 be supported.

725 **Table 11 – Operations: CIM\_HostedAccessPoint**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None

Operation	Requirement	Messages
References	Unspecified	None
ReferenceNames	Unspecified	None

## 726 8.10 CIM\_HostedService

727 Table 12 lists operations that either have special requirements beyond those from [DSP0200](#) or shall not  
728 be supported.

729 **Table 12 - Operations: CIM\_HostedService**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

## 730 8.11 CIM\_ProtocolService

731 Table 13 lists implementation requirements for operations. If implemented, these operations shall be  
732 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 13, all operations  
733 in the default list in 8.4 shall be implemented as defined in [DSP0200](#).

734 NOTE Related profiles may define additional requirements on operations for the profile class.

735 **Table 13 - Operations: CIM\_ProtocolService**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.11.1.	None

### 736 8.11.1 CIM\_ProtocolService — ModifyInstance

737 When the ElementNameEditSupported property of the CIM\_DNSServiceCapabilities instance has a value  
738 of TRUE, the ModifyInstance operation shall allow the value of the ElementName property of the  
739 CIM\_ProtocolService instance to be modified. The ModifyInstance operation shall enforce the length  
740 restriction specified in the MaxElementNameLen property of the CIM\_DNSServiceCapabilities instance.  
741 When the ElementNameEditSupported property of the CIM\_DNSServiceCapabilities has a value of  
742 FALSE, the ModifyInstance operation shall not change the value of the ElementName property of the  
743 CIM\_ProtocolService instance.

## 744 8.12 CIM\_ProvidesEndpoint

745 Table 14 lists implementation requirements for operations. If implemented, these operations shall be  
746 implemented as defined in [in DSP0200](#). In addition, and unless otherwise stated in Table 14, all operations  
747 in the default list in 8.4 shall be implemented as defined in [DSP0200](#).

748 **Table 14 - Operations: CIM\_ProvidesEndpoint**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

749 **8.13 CIM\_ServiceAccessBySAP**

750 Table 15 lists implementation requirements for operations. If implemented, these operations shall be  
 751 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 15, all operations  
 752 in the default list in 8.4 shall be implemented as defined in [DSP0200](#).

753 **Table 15 - Operations: CIM\_ServiceAccessBySAP**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

754 **8.14 CIM\_IPProtocolEndpoint**

755 All operations in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

756 **8.15 CIM\_TCPProtocolEndpoint**

757 Table 16 lists implementation requirements for operations. If implemented, these operations shall be  
 758 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 16, all operations  
 759 in the default list in 8.4 shall be implemented as defined in [DSP0200](#).

760 **Table 16 - Operations: CIM\_TCPProtocolEndpoint**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.16.1.	None
DeleteInstance	Optional. See 8.15.2.	None

761 **8.15.1 ModifyInstance**

762 The ModifyInstance operation may be supported for an instance of CIM\_TCPProtocolEndpoint. When the  
 763 ModifyInstance operation is supported for an instance of CIM\_TCPProtocolEndpoint, the ModifyInstance  
 764 operation shall not modify the following properties:

- 765 • NameFormat
- 766 • ProtocolType

- 767       • PortNumber

### 768   **8.15.2 DeletelInstance**

769   When the CIM\_ProtocolService.ListenOnPort( ) method is supported for the instance of  
770   CIM\_ProtocolService with which the CIM\_TCPProtocolEndpoint is associated through an instance of  
771   CIM\_ServiceAccessBySAP, the DeletelInstance operation shall be supported for the instance of  
772   CIM\_TCPProtocolEndpoint. When the CIM\_ProtocolService.ListenOnPort( ) method is not supported, the  
773   DeletelInstance operation shall not be supported.

774   When the DeletelInstance operation is successful for an instance of CIM\_TCPProtocolEndpoint, the DNS  
775   service shall stop listening on the TCP/IP port indicated by the PortNumber property of the  
776   CIM\_TCPProtocolEndpoint. The implementation shall also remove any association instances that  
777   reference the CIM\_TCPProtocolEndpoint.

### 778   **8.16 CIM\_DNSServiceProtocolEndpoint**

779   Table 17 lists implementation requirements for operations. If implemented, these operations shall be  
780   implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 17, all operations  
781   in the default list in 8.4 shall be implemented as defined in [DSP0200](#).

782   NOTE Related profiles may define additional requirements on operations for the profile class.

783                   **Table 17 – Operations: CIM\_DNSServiceProtocolEndpoint**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.16.1.	None
DeletelInstance	Optional. See 8.16.2.	None

#### 784   **8.16.1 ModifyInstance**

785   The ModifyInstance operation may be supported for an instance of CIM\_DNSServiceProtocolEndpoint.  
786   When the ModifyInstance operation is supported for an instance of CIM\_DNSServiceProtocolEndpoint,  
787   the ModifyInstance operation shall not modify the following properties:

- 788       • NameFormat  
789       • ProtocolIFTType  
790       • OtherTypeDescription

#### 791   **8.16.2 DeletelInstance**

792   The DeletelInstance operation may be supported for instances of CIM\_DNSServiceProtocolEndpoint.  
793   When the DeletelInstance operation is invoked against an instance, the corresponding DNS resolution  
794   session shall be terminated prior to deleting the CIM\_DNSServiceProtocolEndpoint instance. The  
795   implementation shall also remove any association instances that reference the  
796   CIM\_DNSServiceProtocolEndpoint.

### 797   **8.17 CIM\_UDPProtocolEndpoint**

798   Table 18 lists implementation requirements for operations. If implemented, these operations shall be  
799   implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 18, all operations  
800   in the default list in 8.4 shall be implemented as defined in [DSP0200](#).

801

**Table 18 – Operations: CIM\_UDPProtocolEndpoint**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.17.1.	None
DeleteInstance	Optional. See 8.17.2.	

802 **8.17.1 ModifyInstance**

803 The ModifyInstance operation may be supported for an instance of CIM\_UDPProtocolEndpoint. When the  
 804 ModifyInstance operation is supported for an instance of CIM\_UDPProtocolEndpoint, the ModifyInstance  
 805 operation shall not modify the following properties:

- 806 • NameFormat
- 807 • ProtocolIFTType
- 808 • PortNumber

809 **8.17.2 DeleteInstance**

810 When the CIM\_ProtocolService.ListenOnUDPPort( ) method is supported for the instance of  
 811 CIM\_ProtocolService with which the CIM\_UDPProtocolEndpoint is associated through an instance of  
 812 CIM\_ServiceAccessBySAP, the DeleteInstance operation shall be supported for the instance of  
 813 CIM\_UDPProtocolEndpoint. When the CIM\_ProtocolService.ListenOnUDPPort( ) method is not  
 814 supported, the DeleteInstance operation should not be supported.

815 When the DeleteInstance operation is successful for an instance of CIM\_UDPProtocolEndpoint, the DNS  
 816 service shall stop listening on the UDP port indicated by the PortNumber property of the  
 817 CIM\_UDPProtocolEndpoint. The implementation shall also remove any association instances that  
 818 reference the CIM\_UDPProtocolEndpoint.

819 **8.18 CIM\_DNSServiceCapabilities**

820 All operations in the default list in 8.4 shall be implemented as defined in [DSP0200](#).

821 NOTE Related profiles may define additional requirements on operations for the profile class.

822 **8.19 CIM\_DNSServiceSettingData**

823 Table 19 lists implementation requirements for operations. If implemented, these operations shall be  
 824 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 19, all operations  
 825 in the default list in 8.4 shall be implemented as defined in [DSP0200](#).

826 NOTE Related profiles may define additional requirements on operations for the profile class.

827 **Table 19 – Operations: CIM\_DNSServiceSettingData**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.19.1.	None

828 **8.19.1 CIM\_DNSServiceSettingData — ModifyInstance**

829 When the CIM\_DNSServerSettingData instance is associated with the CIM\_ProtocolService instance  
 830 through an instance of CIM\_ElementSettingData and the value of the IsDefault property of the

831 CIM\_ElementSettingData instance has a value of 1 (Is Default), the ModifyInstance operation shall not be  
832 supported.

833 When the CIM\_DNServerSettingData instance is not associated with an instance of CIM\_ProtocolService  
834 through an instance of CIM\_ElementSettingData where the IsDefault property has a value of 1 (Is  
835 Default), the ModifyInstance operation may be supported for the CIM\_DNServerSettingData instance.

## 836 9 Use cases

837 This clause contains object diagrams and use cases for the *DNS Service Management Profile*.

### 838 9.1 Profile Registration

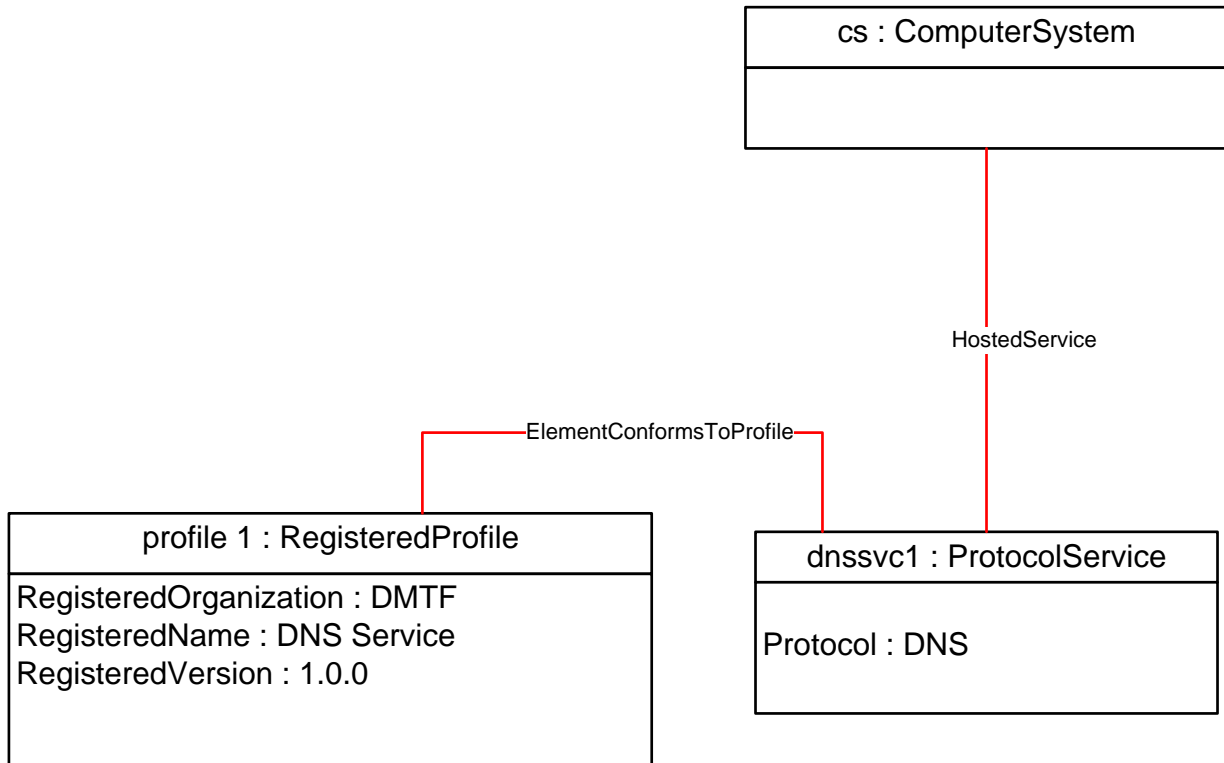
839 The object diagrams in Figure 2 and Figure 3 show two possible methods for advertising profile  
840 conformance.

841 The object diagram in Figure 2 describes method for advertising profile conformance using  
842 CIM\_ComputerSystem. The instances of CIM\_RegisteredProfile are used to identify the version of the  
843 *DNS Service Management Profile* with which an instance of CIM\_ProtocolService is conformant. An  
844 instance of CIM\_RegisteredProfile exists for each profile that is instrumented in the system. The  
845 CIM\_ProtocolService instance is scoped to an instance of CIM\_ComputerSystem. **NOTE: Both the**  
846 **highlighted text above and the diagram below are incorrect. For example the Base Server profile on the**  
847 **diagram is not mentioned anywhere in the profile.**

848 **[Ghazanfar: There are two choices to reflect profile registration. One of the options will be selected based**  
849 **on WG consensus.]**

850 **Meeting: Keep both options.**

851

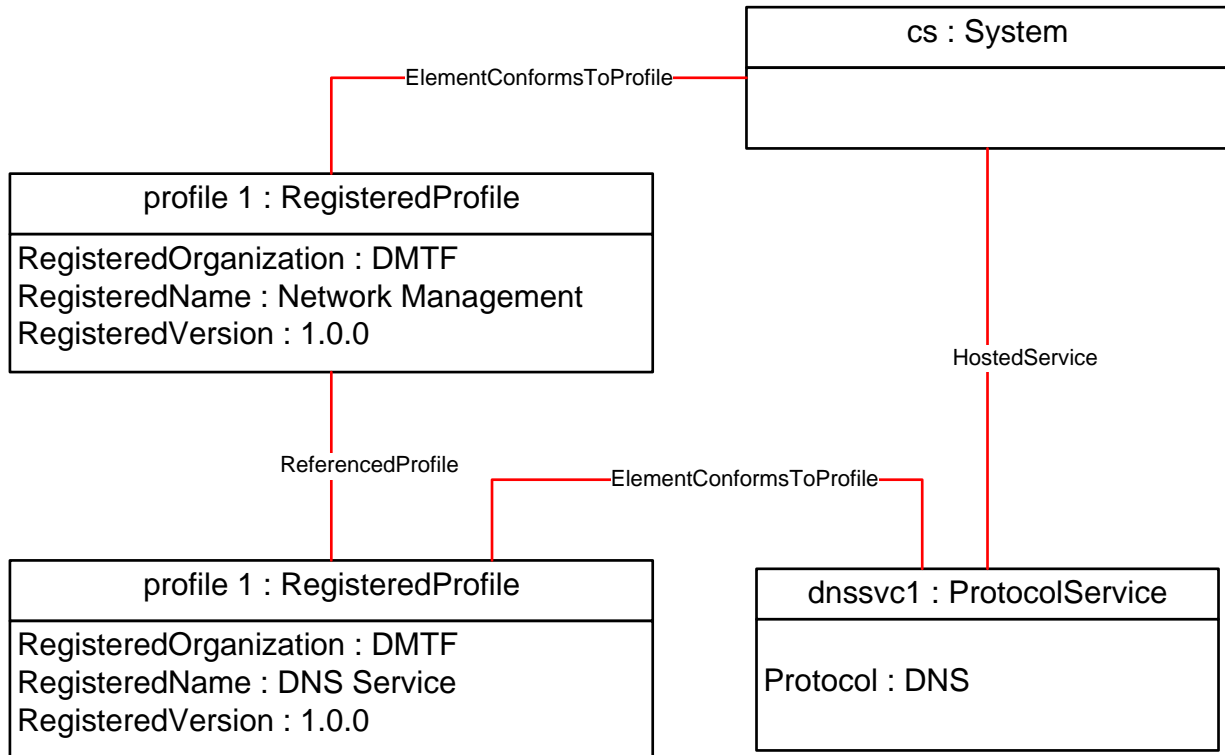


852

853

**Figure 2 – Registered profile (ComputerSystem)**

854 The object diagram in Figure 3 describes method for advertising profile conformance using CIM\_System  
 855 (i.e. network management). The instances of CIM\_RegisteredProfile are used to identify the version of  
 856 the *DNS Service Management Profile* with which an instance of CIM\_ProtocolService is conformant. An  
 857 instance of CIM\_RegisteredProfile exists for each profile that is instrumented in the system. One instance  
 858 of CIM\_RegisteredProfile identifies the System conforming to the Network Management Profile. The other  
 859 instance identifies an instance of CIM\_ProtocolService. The CIM\_ProtocolService instance is scoped to  
 860 an instance of CIM\_System. This instance of CIM\_System is conformant with the DMTF Network  
 861 Management Profile version 1.0.0 as indicated by the CIM\_ElementConformsToProfile association to the  
 862 CIM\_RegisteredProfile instance.



863

864

**Figure 3 - Registered profile (Network Management)**

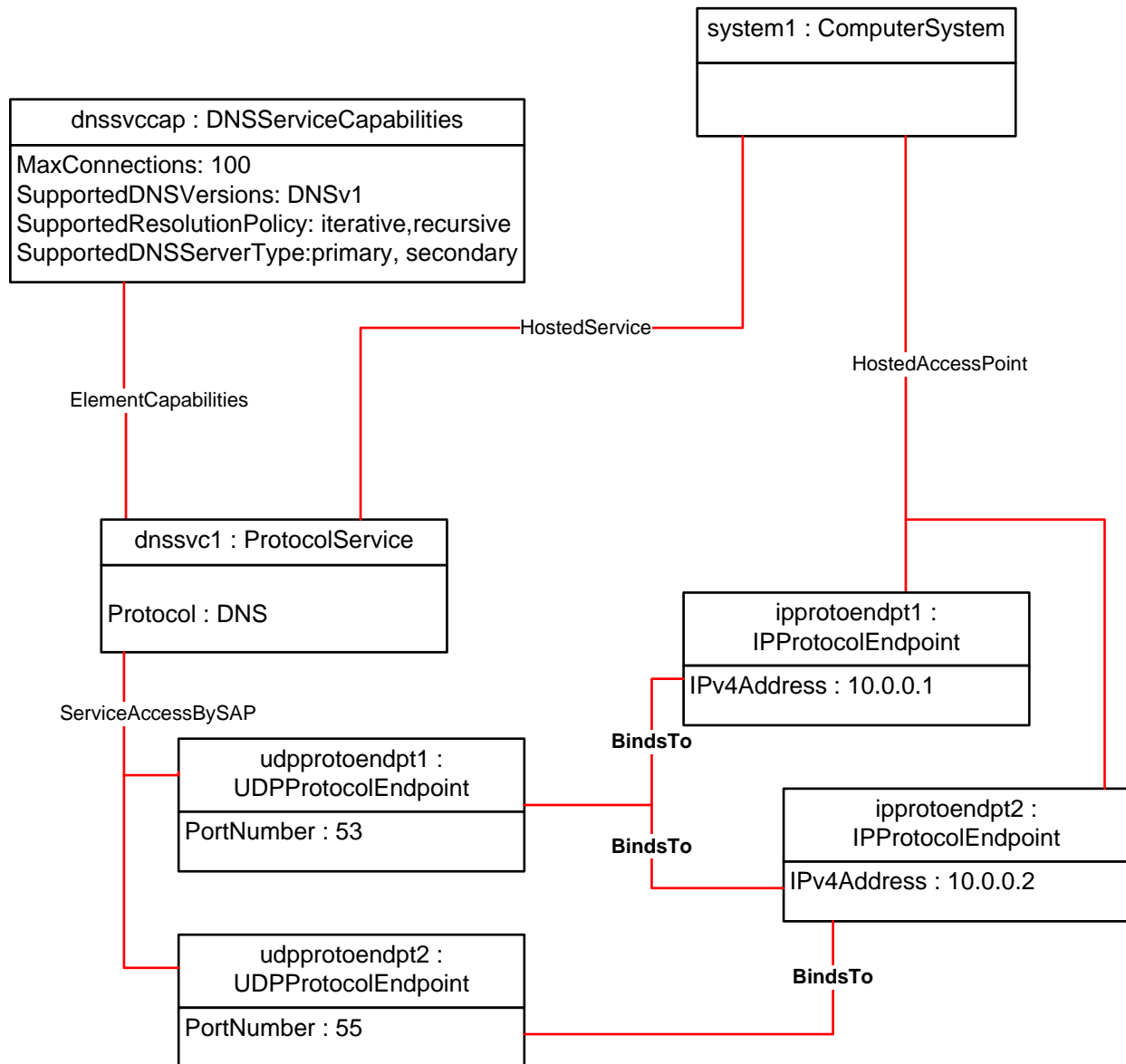
865 **9.2 Adding a UDP port for the DNS service**

866 An implementation can support adding and removing bindings between the DNS service and UDP ports.  
 867 When an implementation supports adding bindings, a client can configure the service to be accessible on  
 868 all interfaces or a specific interface.

869 To have the DNS service accessible on a UDP port across all IP interfaces of the system, the client can  
 870 invoke the ListenOnUDPPort() method of the CIM\_ProtocolService instance, specifying the desired  
 871 PortNumber.

872 To have the DNS service accessible on a UDP port for a specific interface, the client can invoke the  
 873 ListenOnUDPPort( ) method of the CIM\_ProtocolService instance, specifying a reference to the  
 874 CIM\_IPProtocolEndpoint instance that represents the specific IP interface.





875

876

**Figure 4 - UDP port configuration to specific interface**

877

Figure 4 reflects the configuration where the ListenOnUDPPort( ) method was invoked with the IPEndpoint parameter containing a reference to ipprotoendpt2 and a PortNumber parameter of 55. The instance udpproendpt2 is created and associated with ipprotoendpt2.

878

879

### 880 9.3 Determining the DNS service capabilities

881

A user can determine the capabilities of the DNS service as follows:

- 882 1) Find the instance of CIM\_DNSServiceCapabilities associated with the CIM\_ProtocolService  
883 through an instance of CIM\_ElementCapabilities.
- 884 2) View the properties of the CIM\_DNSServiceCapabilities instance to see the supported  
885 functions.

### 886 9.3.1 Determine which IP address versions are supported

887 View the DNSDomainType property of the CIM\_DNSServiceCapabilities instance to determine the  
888 support for IPv4 (IN-ADDR.ARPA) and IPv6(IP6.ARPA) addresses.

889 IN-ADDR.ARPA property represents a domain that is defined to look up a record given an IPv4 address.  
890 In addition, IP6.ARPA property represents a special domain that is defined to look up a record given an  
891 IPv6 address.

## 892 9.4 Configuring DNS service default settings

893 When a DNS resolution session is established, DNS resolution session settings have default values. A  
894 client can change the default values for subsequent sessions' settings as follows:

- 895 1) Find the instance of CIM\_ElementSettingData that associates an instance of  
896 CIM\_DNSServiceSettingData with the CIM\_ProtocolService, where the value of the IsDefault  
897 property is 1 (Is Default).
- 898 2) Modify the properties of the referenced CIM\_DNSServiceSettingData instance.

## 899 9.5 Modifying DNS service active settings

900 Clients can find the active DNS resolution sessions for a DNS service and modify their configuration as  
901 follows:

- 902 1) From the instance of CIM\_DNSServiceProtocolEndpoint, find the associated instance of the  
903 CIM\_DNSServiceSettingData through the CIM\_ElementSettingData association, where the  
904 value of the IsCurrent property is 1.
- 905 2) Modify the properties of the referenced CIM\_DNSServiceSettingData instance.

### 906 9.5.1 Enabling DNS service security policy

907 Transaction Signature (TSIG) provides mechanisms for protecting a message from a primary to  
908 secondary DNS server and vice versa ([RFC2845](#)).

909 DNS SEC ([RFC 4033](#), [RFC4034](#), and [RFC4035](#)) provides mechanisms to establish authenticity and  
910 integrity of DNS data.

911 A client can activate TSIG and/or DNS SEC policies as follows:

- 912 1) From the instance of CIM\_DNSServiceProtocolEndpoint, find the associated instance of the  
913 CIM\_DNSServiceSettingData through the CIM\_ElementSettingData association, where the  
914 value of the IsCurrent property is 1.
- 915 2) Set the values of TSIGEnable and/or DNSSEcEnable properties of  
916 CIM\_DNSServiceSettingData instance equal to 1.

## 917 9.6 Viewing DNS service active settings

918 A client can view the active configuration for DNS server as follows:

- 919 1) From the instance of CIM\_DNSServiceProtocolEndpoint, find the associated instance of the
- 920 CIM\_DNSServiceSettingData through the CIM\_ElementSettingData association, where the
- 921 value of the IsCurrent property is 1.
- 922 2) Instance of the CIM\_DNSServiceSettingData represents the active settings..

923 **9.6.1 View the DNS service IP address**

924 A user can find the DNS service IP address as follows:

- 925 1) Find the instance of CIM\_DNSServiceProtocolEndpoint associated with either the
- 926 CIM\_UDPProtocolEndpoint or CIM\_TCPProtocolEndpoint through an instance of CIM\_BindsTo.
- 927 2) Find the instance of CIM\_IPProtocolEndpoint associated with either the
- 928 CIM\_UDPProtocolEndpoint or CIM\_TCPProtocolEndpoint through an instance of CIM\_BindsTo.
- 929 3) View the IPv4Address and IPv6Address properties of the CIM\_IPProtocolEndpoint instance to
- 930 find the IP address of the DNS service.

931 **9.6.2 View the DNS resolution session policy**

932 A user can determine the DNS resolution session policy of the DNS service as follows:

- 933 1) From the instance of CIM\_DNSServiceProtocolEndpoint, find the associated instance of the
- 934 CIM\_DNSServiceSettingData through the CIM\_ElementSettingData association, where the
- 935 value of the IsCurrent property is 1.
- 936 2) View the value of DNSResolutionPolicy property of the CIM\_DNSServiceSettingData instance
- 937 to find the DNS resolution session policy of the DNS service.

938 **9.7 Stopping the DNS service on a specific UDP port**

939 A management client can stop the DNS service from being accessible on a specific UDP port by invoking

940 the intrinsic DeleteInstance operation against the instance of CIM\_UDPProtocolEndpoint that represents

941 the UDP port.

942 Using the configuration shown in Figure 4 as an example, invoking the DeleteInstance operation against

943 the instance udpprotoendpt2 would cause the DNS service to be no longer accessible on port 55.

944 **9.8 Disabling the DNS service**

945 If an implementation supports disabling the DNS service, a user can disable the DNS service by invoking

946 the RequestStateChange( ) method on CIM\_ProtocolService instance with a value of Disabled for the

947 RequestedState parameter.

948 **10 CIM Elements**

949 Table 20 shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be

950 implemented as described in Table 20. Clauses 7 (“Implementation”) and 8 (“Methods”) may impose

951 additional requirements on these elements.

952 **Table 20 - CIM Elements: DNS service management profile**

Element Name	Requirement	Description
<b>Classes</b>		

Element Name	Requirement	Description
CIM_BindsTo	Conditional	See clauses 10.1, 10.2, and 10.3
CIM_DNSServiceCapabilities	Mandatory	See clause 10.4
CIM_DNSServiceProtocolEndpoint	Mandatory	See clause 10.6
CIM_DNSServiceProtocolEndpointStats	Mandatory	See clause 10.8
CIM_DNSServiceSettingData	Mandatory	See clause 10.5
CIM_ElementCapabilities	Optional	See clause 10.16
CIM_ElementConformsToProfile	Optional	See clause 10.20
CIM_ElementSettingData	Optional	See clauses 10.9 and 10.10
CIM_HostedAccessPoint	Conditional	See clauses 10.11 and 10.12
CIM_HostedService	Conditional	See clause 10.13
CIM_IPProtocolEndpoint	Optional	See clause 10.14
CIM_ProvidesEndpoint	Mandatory	See clause 10.17
CIM_ProtocolService	Conditional	See clause 10.7
CIM_RegisteredProfile	Optional	See clause 10.15
CIM_ServiceAccessBySAP	Conditional	See clauses 10.18 and 10.19
CIM_TCPProtocolEndpoint	Optional	See clause 10.21
CIM_UDPProtocolEndpoint	Mandatory	See clause 10.22

953 **10.1 CIM\_BindsTo — CIM\_IPProtocolEndpoint and CIM\_TCPProtocolEndpoint**

954 CIM\_BindsTo relates the CIM\_IPProtocolEndpoint instance with the CIM\_TCPProtocolEndpoint instance  
 955 on which it depends. Table 21 provides information about the properties of CIM\_BindsTo.

956 **Table 21 - Class: CIM\_BindsTo — CIM\_IPProtocolEndpoint and CIM\_TCPProtocolEndpoint**

Elements	Requirement	Description
Antecedent	Mandatory	<b>Key:</b> This shall be a reference to an instance of CIM_TCPProtocolEndpoint. Cardinality 1..*
Dependent	Mandatory	<b>Key:</b> This shall be a reference to an instance of CIM_IPProtocolEndpoint. Cardinality *

957 **10.2 CIM\_BindsTo — CIM\_IPProtocolEndpoint and CIM\_UDPProtocolEndpoint**

958 CIM\_BindsTo relates the CIM\_IPProtocolEndpoint instance with the CIM\_UDPProtocolEndpoint instance  
 959 on which it depends. Table 22 provides information about the properties of CIM\_BindsTo.

960 **Table 22 - Class: CIM\_BindsTo – CIM\_IPProtocolEndpoint and CIM\_UDPProtocolEndpoint**

Elements	Requirement	Description
Antecedent	Mandatory	<b>Key:</b> This shall be a reference to an instance of CIM_UDPProtocolEndpoint. Cardinality 1..*
Dependent	Mandatory	<b>Key:</b> This shall be a reference to an instance of CIM_IPProtocolEndpoint. Cardinality *

961 **10.3 CIM\_BindsTo — CIM\_TCPProtocolEndpoint and**  
 962 **CIM\_DNSServiceProtocolEndpoint**

963 CIM\_BindsTo relates the CIM\_TCPProtocolEndpoint instance with the CIM\_DNSServiceProtocolEndpoint  
 964 instance on which it depends. Table 23 provides information about the properties of CIM\_BindsTo.

965 **Table 23 - Class: CIM\_BindsTo – CIM\_TCPProtocolEndpoint and**  
 966 **CIM\_DNSServiceProtocolEndpoint**

Elements	Requirement	Description
Antecedent	Mandatory	<b>Key:</b> This shall be a reference to an instance of CIM_DNSServiceProtocolEndpoint. Cardinality 1..*
Dependent	Mandatory	<b>Key:</b> This shall be a reference to an instance of CIM_TCPProtocolEndpoint. Cardinality *

967 **10.4 CIM\_DNSServiceCapabilities**

968 CIM\_DNSServiceCapabilities represents the capabilities of a DNS service. Table 24 provides information  
 969 about the properties of CIM\_DNSServiceCapabilities.

970 **Table 24 - Class: CIM\_DNSServiceCapabilities**

Elements	Requirement	Description
InstanceID	Mandatory	Key
SupportedDNSDomainTypes [ ]	Mandatory	This property shall have a value of 1 (IPv4/ IN-ADDR.ARPA) , 2 (IPv6/IP6.ARPA) or both.
ReverseResolutionSupport	Optional	This property shall have a boolean value i.e., true indicates support and false indicates feature not supported.

Elements	Requirement	Description
SupportedNameServerTypes [ ]	Mandatory	This property indicates role of DNS name server and shall have a value of 1 (Primary name server), (Secondary name server), 3 (Caching-only name server), or all.
RecursionSupport	Optional	This property shall have a boolean value. true means the server deals with DNS queries according to recursive mechanisms and false means all queries are processed nonrecursively.
DNSNotifySupport	Optional	The DNSNotify operation is described in RFC1996. DNS Notify can inform the secondary name servers about data changes in the zone. This property shall have a boolean value i.e., true indicates support and false indicates feature not supported.
SupportedDNSwks	Mandatory	A description of well-known server services (wks) in TCP and UDP. It consists of three parts: 32-bit address, protocol number, and service ports.
DNSUpdateSupport	Optional	The DNS Update mechanism is described in RFC3007. The DNS Update operation enables dynamic correction of entries in the DNS database. This property shall have a boolean value i.e., true indicates support and false indicates feature not supported.
TSIGSupport	Optional	This property indicates support for TSIG. This property shall have a boolean value i.e., true indicates support and false indicates feature not supported.
DNSsecSupport	Mandatory	This property indicates support for DNSSEC. This property shall have a boolean value i.e., true indicates support and false indicates feature not supported.
QuerySupport	Optional	This parameter indicates the support for query. This property shall have a boolean value i.e., true indicates support and false indicates feature not supported.
ZoneTransferSupport	Optional	This property indicates support of zone transfer capability. This property shall have a boolean value i.e., true indicates support and false indicates feature not supported.
EDNS0Support	Optional	This property indicates support of EDNS0 capability. This property shall have a boolean value i.e., true indicates support and false indicates feature not supported.

971 **10.5 CIM\_DNSServiceSettingData**

972 CIM\_DNSServiceSettingData represents the settings for the DNS server. Table 25 provides information  
 973 about the properties of CIM\_DNSServiceSettingData.

974 **Table 25 - Class: CIM\_DNSServiceSettingData**

Elements	Requirement	Description
InstanceID	Mandatory	Key
DNSDomainTypes [ ]	Mandatory	This property shall have a value of 1 (IPv4/ IN-ADDR.ARPA), 2 (IPv6/IP6.ARPA) or both.

Elements	Requirement	Description
EnableReverseResolution	Optional	This property shall have a boolean value i.e., true enable reverse resolution and false disable reverse resolution .
NameServerType []	Mandatory	This property indicates role of DNS name server and shall have a value of 1 (Primary name server), (Secondary name server), or 3 (Caching-only name server).
EnableRecursion	Optional	This property shall have a boolean value, i.e., true means the server deals with DNS queries according to recursive mechanisms and false means all queries are processed nonrecursively.
EnableDNSNotify	Optional	The DNSNotify operation is described in RFC1996. DNS Notify can inform the secondary name servers about data changes in the zone. This property shall have a boolean value i.e., true enables DNSNotify and false disables DNSNotify.
DNSwks	Mandatory	A description of well-known server services (wks) in TCP and UDP. It consists of three parts: 32-bit address, protocol number, and service ports.
EnableDNSUpdate	Optional	The DNS Update mechanism is described in RFC3007. The DNS Update operation enables dynamic correction of entries in the DNS database. This property shall have a boolean value i.e., true enables DNSUpdate and false disables DNSUpdate.
EnableTSIG	Optional	This property is used to enable TSIG. This property shall have a boolean value i.e., true enables EnableTSIG and false disables EnableTSIG.
EnableDNSsec	Mandatory	This property is used to enable DNSSEC. This property shall have a boolean value i.e., true enables DNSSec and false disables DNSSec.
AllowQuery	Optional	This parameter defines which hosts are allowed to ask a query. This property shall have a boolean value i.e., true enables AllowQuery and false disables AllowQuery.
AllowTransfer	Optional	This property defines which hosts are permitted to do a zone transfer from the server. This property shall have a boolean value i.e., true enables AllowTransfer and false disables AllowTransfer.
TTL	Optional	This property specifies the time interval that the resource record may be cached before the source of the information should again be consulted.
EnableEDNS0	Optional	<a href="#">[RFC2671]</a> extension to DNS to allow for the larger packets. This property shall have a boolean value i.e., true enables EDNS0 and false disables EDNS0.

975 **10.6 CIM\_DNSServiceProtocolEndpoint**

976 CIM\_DNSServiceProtocolEndpoint represents the DNS server on the system. Table 26 provides  
 977 information about the properties of CIM\_DNSServiceProtocolEndpoint.

Table 26 – Class: CIM\_DNSServiceProtocolEndpoint

Elements	Requirement	Description
SystemCreationClassName	Mandatory	Key
CreationClassName	Mandatory	Key
SystemName	Mandatory	Key
Name	Mandatory	Key
ProtocolIFType	Mandatory	This property shall have a value of 1 (“Other”).
OtherTypeDescription	Mandatory	This property shall have a value of “DNS”.
DNSDomainTypes []	Mandatory	This property shall have a value of 1 (IPv4/ IN-ADDR.ARPA), 2 (IPv6/IP6.ARPA) or both.
EnableReverseResolution	Optional	This property shall have a boolean value i.e., true enable reverse resolution and false disable reverse resolution .
NameServerType []	Mandatory	This property indicates role of DNS name server and shall have a value of 1 (Primary name server), (Secondary name server), or 3 (Caching-only name server).
EnableRecursion	Optional	This property shall have a boolean value, i.e., true means the server deals with DNS queries according to recursive mechanisms and false means all queries are processed nonrecursively.
EnableDNSNotify	Optional	The DNSNotify operation is described in RFC1996. DNS Notify can inform the secondary name servers about data changes in the zone. This property shall have a boolean value i.e., true enables DNSNotify and false disables DNSNotify.
DNSwks	Mandatory	A description of well-known server services (wks) in TCP and UDP. It consists of three parts: 32-bit address, protocol number, and service ports.
EnableDNSUpdate	Optional	The DNS Update mechanism is described in RFC3007. The DNS Update operation enables dynamic correction of entries in the DNS database. This property shall have a boolean value i.e., true enables DNSUpdate and false disables DNSUpdate.
EnableTSIG	Optional	This property is used to enable TSIG. This property shall have a boolean value i.e., true enables EnableTSIG and false disables EnableTSIG.
EnableDNSsec	Mandatory	This property is used to enable DNSSEC. This property shall have a boolean value i.e., true enables DNSSec and false disables DNSSec.
AllowQuery	Optional	This parameter defines which hosts are allowed to ask a query. This property shall have a boolean value i.e., true enables AllowQuery and false disables AllowQuery.



Elements	Requirement	Description
AllowTransfer	Optional	This property defines which hosts are permitted to do a zone transfer from the server. This property shall have a boolean value i.e., true enables AllowTransfer and false disables AllowTransfer.
TTL	Optional	This property specifies the time interval that the resource record may be cached before the source of the information should again be consulted.
EnableEDNS0	Optional	[RFC2671] extension to DNS to allow for the larger packets. This property shall have a boolean value i.e., true enables EDNS0 and false disables EDNS0.

979

980 **10.7 CIM\_ProtocolService**

981 CIM\_ProtocolService represents the DNS service. Table 27 provides information about the properties of  
 982 CIM\_ProtocolService.

983

**Table 27 - Class: CIM\_ProtocolService**

Elements	Requirement	Description
SystemCreationClassName	Mandatory	None
CreationClassName	Mandatory	None
SystemName	Mandatory	None
Name	Mandatory	None
Protocol	Mandatory	See 7.1.1
MaxConnections	Mandatory	A value of 0 (zero) shall indicate unknown.
RequestedState	Mandatory	See 7.1.1
EnabledState	Mandatory	See 7.1.1
ElementName	Mandatory	None
OperationalStatus	Mandatory	None
HealthState	Mandatory	None
RequestStateChange( )	Mandatory	See 8.3.
ListenOnPort( )	Conditional	See 8.2.
ListenOnUDPPort( )	Conditional	See 8.3

984 **10.8 CIM\_DNSServiceProtocolEndpointStats**

985 The CIM\_DNSServiceProtocolEndpointStats represents the statistics of the DNS service that includes  
 986 total number of resolution requests to DNS server, the total number of requests resolved locally/by server

987 itself, the total number of requests resolved by support of other DNS server(s), usage frequency of  
 988 Domain Names, and so on.

989 Table 28 provides information about the properties of CIM\_DNSServiceProtocolEndpointStats.

990 **Table 28 - Class: CIM\_DNSServiceProtocolEndpointStats**

Elements	Requirement	Description
ResolutionRequests	Mandatory	This property indicates total number of resolutions requests received.
RequestsResolvedLocally	Mandatory	This property indicates number of requests resolved locally.
RequestsResolvedRemotely	Conditional	This property indicates number of requests resolved via remote DNS servers.

991 **10.9 CIM\_ElementSettingData — CIM\_ProtocolService and**  
 992 **CIM\_DNSServiceSettingData**

993 CIM\_ElementSettingData associates instances of CIM\_DNSServiceSettingData with the  
 994 CIM\_ProtocolService instance. Table 29 provides information about the properties of  
 995 CIM\_ElementSettingData.

996 **Table 29 - Class: CIM\_ElementSettingData — CIM\_ProtocolService and**  
 997 **CIM\_DNSServiceSettingData**

Elements	Requirement	Description
ManagedElement	Mandatory	<b>Key:</b> This shall be a reference to the Central Instance. Cardinality 1
SettingData	Mandatory	<b>Key:</b> This shall be a reference to an instance of CIM_DNSServiceSettingData. Cardinality *

998 **10.10 CIM\_ElementSettingData — CIM\_DNSServiceProtocolEndpoint and**  
 999 **CIM\_DNSServiceSettingData**

1000 CIM\_ElementSettingData associates instances of CIM\_DNSServiceSettingData with the  
 1001 CIM\_DNSServiceProtocolEndpoint instance. Table 30 provides information about the properties of  
 1002 CIM\_ElementSettingData.

1003 **Table 30 - Class: CIM\_ElementSettingData — CIM\_DNSServiceProtocolEndpoint and**  
 1004 **CIM\_DNSServiceSettingData**

Elements	Requirement	Description
ManagedElement	Mandatory	<b>Key:</b> This shall be a reference to the Central Instance. Cardinality *
SettingData	Mandatory	<b>Key:</b> This shall be a reference to an instance of CIM_DNSServiceSettingData. Cardinality 1..*

1005 **10.11 CIM\_HostedAccessPoint — CIM\_System and CIM\_IPProtocolEndpoint**

1006 CIM\_HostedAccessPoint associates an instance of CIM\_IPProtocolEndpoint with scoping CIM\_System.  
 1007 Table 31 provides information about the properties of CIM\_HostedAccessPoint.

1008 **Table 31 - Class: CIM\_HostedAccessPoint — CIM\_System and CIM\_IPProtocolEndpoint**

Elements	Requirement	Description
Antecedent	Mandatory	<b>Key:</b> This shall be a reference to the Scoping Instance. Cardinality 1
Dependent	Mandatory	<b>Key:</b> This shall be a reference to an instance of CIM_IPProtocolEndpoint Cardinality *

1009 **10.12 CIM\_HostedAccessPoint — CIM\_System and**  
 1010 **CIM\_DNSServiceProtocolEndpoint**

1011 CIM\_HostedAccessPoint associates an instance of CIM\_DNSServiceProtocolEndpoint with scoping  
 1012 CIM\_System. Table 32 provides information about the properties of CIM\_HostedAccessPoint.

1013  
1014**Table 32 - Class: CIM\_HostedAccessPoint — CIM\_System and  
CIM\_DNSServiceProtocolEndpoint**

Elements	Requirement	Description
Antecedent	Mandatory	<b>Key:</b> This shall be a reference to the Scoping Instance. Cardinality 1
Dependent	Mandatory	<b>Key:</b> This shall be a reference to an instance of CIM_DNSServiceProtocolEndpoint. Cardinality *

1015 **10.13 CIM\_HostedService**

1016 CIM\_HostedService relates the CIM\_ProtocolService instance to its scoping CIM\_System instance. Table  
1017 33 provides information about the properties of CIM\_HostedService.

1018

**Table 33 - Class: CIM\_HostedService**

Elements	Requirement	Description
Antecedent	Mandatory	<b>Key:</b> This shall be a reference to the Scoping Instance. Cardinality 1
Dependent	Mandatory	<b>Key:</b> This shall be a reference to an instance of CIM_ProtocolService. Cardinality *

1019 **10.14 CIM\_IPProtocolEndpoint**

1020 CIM\_IPProtocolEndpoint represents an IP interface that is associated with an Ethernet interface. Table 34  
1021 provides information about the properties of CIM\_IPProtocolEndpoint.

1022

**Table 34 - Class: CIM\_IPProtocolEndpoint**

Elements	Requirement	Description
SystemCreationClassName	Mandatory	<b>Key</b>
CreationClassName	Mandatory	<b>Key</b>
SystemName	Mandatory	<b>Key</b>
Name	Mandatory	<b>Key</b>
NameFormat	Mandatory	Pattern ".*"
ProtocolIFType	Mandatory	See.
ElementName	Mandatory	Pattern ".*"
IPv4Address	Conditional	None.
SubnetMask	Conditional	None.
AddressOrigin	Mandatory	None.

Elements	Requirement	Description
IPv6Address	Conditional	None.
IPv6SubnetPrefixLength	Conditional	None.

1023 **10.15 CIM\_RegisteredProfile**

1024 CIM\_RegisteredProfile identifies the *DNS Service Management Profile* in order for a client to determine  
 1025 whether an instance of CIM\_IPProtocolEndpoint is conformant with this profile. The  
 1026 CIM\_RegisteredProfile class is defined by the *Profile Registration Profile* ([DSP1033](#)). With the exception  
 1027 of the mandatory values specified for the properties in Table 35, the behavior of the  
 1028 CIM\_RegisteredProfile instance is in accordance with [DSP1033](#).

1029 **Table 35 - Class: CIM\_RegisteredProfile**

Elements	Requirement	Description
RegisteredName	Mandatory	This property shall have a value of "DNS Service".
RegisteredVersion	Mandatory	This property shall have a value of "1.0.0".
RegisteredOrganization	Mandatory	This property shall have a value of "DMTF".

1030 **10.16 CIM\_ElementCapabilities**

1031 CIM\_ElementCapabilities is used to associate an instance of CIM\_DNSServiceCapabilities with the  
 1032 CIM\_ProtocolService. Table 36 provides information about the properties of CIM\_ElementCapabilities.

1033 **Table 36 - Class: CIM\_ElementCapabilities**

Elements	Requirement	Description
ManagedElement	Mandatory	This property shall be a reference to the Central Instance. Cardinality 1..*
Capabilities	Mandatory	This property shall be a reference to the CIM_DNSServiceCapabilities instance. Cardinality 1

1034 **10.17 CIM\_ProvidesEndpoint**

1035 CIM\_ProvidesEndpoint is used to associate the instance of CIM\_ProtocolService with an instance of  
 1036 CIM\_DNSServiceProtocolEndpoint that represents a DNS server. Table 37 provides information about  
 1037 the properties of CIM\_ProvidesEndpoint.

1038

**Table 37 - Class: CIM\_ProvidesEndpoint**

Elements	Requirement	Description
Antecedent	Mandatory	This property shall be a reference to the instance of CIM_ProtocolService. Cardinality 1
Dependent	Mandatory	This property shall be a reference to an instance of CIM_DNSServerProtocolEndpoint. Cardinality *

1039

### 10.18 CIM\_ServiceAccessBySAP - CIM\_TCPProtocolEndpoint and CIM\_ProtocolService

1040

1041 CIM\_ServiceAccessBySAP is used to associate the instance of CIM\_ProtocolService with an instance of  
 1042 CIM\_TCPProtocolEndpoint over which a DNS resolution session with the service can be established.  
 1043 Table 38 provides information about the properties of CIM\_ServiceAccessBySAP.

1044

**Table 38 - Class: CIM\_ServiceAccessBySAP - CIM\_TCPProtocolEndpoint and CIM\_ProtocolService**

1045

Elements	Requirement	Description
Antecedent	Mandatory	This property shall be a reference to the instance of CIM_ProtocolService. Cardinality 1..*
Dependent	Mandatory	This property shall be a reference to an instance of CIM_DNSServiceProtocolEndpoint. Cardinality *

1046

### 10.19 CIM\_ServiceAccessBySAP - UDPProtocolEndpoint and CIM\_ProtocolService

1047

1048 CIM\_ServiceAccessBySAP is used to associate the instance of CIM\_ProtocolService with an instance of  
 1049 CIM\_UDPProtocolEndpoint over which a DNS resolution session with the service can be established.  
 1050 Table 39 provides information about the properties of CIM\_ServiceAccessBySAP.

1051

**Table 39 - Class: CIM\_ServiceAccessBySAP - UDPProtocolEndpoint and CIM\_ProtocolService**

Elements	Requirement	Description
Antecedent	Mandatory	This property shall be a reference to the instance of CIM_ProtocolService. Cardinality 1..*
Dependent	Mandatory	This property shall be a reference to an instance of CIM_DNSServiceProtocolEndpoint. Cardinality *

1052 **10.20 CIM\_ElementConformsToProfile**

1053 CIM\_ElementConformsToProfile associates an instance of CIM\_ProtocolService with its corresponding  
 1054 CIM\_RegisteredProfile that represents the version of profile implemented. Table 40 provides information  
 1055 about the properties of CIM\_ElementConformsToProfile.

1056 **Table 40 - Class: CIM\_ElementConformsToProfile**

Elements	Requirement	Description
ManagedElement	Mandatory	<b>Key:</b> This shall be a reference to the instance of CIM_ProtocolService. Cardinality *
ConformantStandard	Mandatory	<b>Key:</b> This shall be a reference to a CIM_RegisteredProfile. Cardinality 1

1057 **10.21 CIM\_TCIPProtocolEndpoint**

1058 CIM\_TCIPProtocolEndpoint represents an IP port to which a DNS service can be bound. Table 41  
 1059 provides information about the properties of CIM\_TCIPProtocolEndpoint.

1060 **Table 41 - Class: CIM\_TCIPProtocolEndpoint**

Elements	Requirement	Description
SystemCreationClassName	Mandatory	None
CreationClassName	Mandatory	None
SystemName	Mandatory	None
Name	Mandatory	None
NameFormat	Mandatory	Pattern ".*"
ProtocolIFType	Mandatory	Matches 4111 ("TCP")
ElementName	Mandatory	Pattern ".*"
PortNumber	Mandatory	None

1061 **10.22 CIM\_UDPPProtocolEndpoint**

1062 CIM\_UDPPProtocolEndpoint represents an IP port to which a DNS service can be bound. Table 42  
 1063 provides information about the properties of CIM\_UDPPProtocolEndpoint.

1064 **Table 42 - Class: CIM\_UDPPProtocolEndpoint**

Elements	Requirement	Description
SystemCreationClassName	Mandatory	None
CreationClassName	Mandatory	None
SystemName	Mandatory	None

Elements	Requirement	Description
Name	Mandatory	None
NameFormat	Mandatory	Pattern ".*"
ProtocolType	Mandatory	Matches 4111 ("UDP")
ElementName	Mandatory	Pattern ".*"
PortNumber	Mandatory	None



**ANNEX A  
(informative)**

**Change log**

Version	Date	Description
1.0.0	2016-02-23	DMTF Work in Progress

1065  
1066  
1067  
1068  
1069

1070  
1071