



Document Identifier: DSP1069

Date: 2015-06-19

Version: 1.0.0b

1  
2  
3  
4

## 5 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 This document is available for public review and comment until superseded.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>

6 **Supersedes:** 1.0.0a  
7 **Document Type:** Specification  
8 **Document Class:** Normative  
9 **Document Status:** Work in Progress  
10 **Document Language:** en-US

11

12 Copyright Notice

13 Copyright © 2015 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28 withdrawn or modified after publication, and shall be indemnified and held harmless by any party  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>.

# CONTENTS

35	Foreword .....	7
36	Introduction.....	8
37	Document conventions.....	8
38	1    Scope .....	9
39	2    Normative references.....	9
40	3    Terms and definitions.....	10
41	4    Symbols and abbreviated terms.....	12
42	5    Synopsis.....	12
43	6    Description .....	13
44	6.1    Class diagram .....	13
45	6.1.1    Support for IP protocol versioning .....	15
46	6.1.2    Management of DNS security.....	15
47	6.1.3    Representation of DNS service usage data .....	15
48	7    Implementation.....	15
49	7.1    Representing a DNS service .....	15
50	7.1.1    CIM_ProtocolService.Protocol.....	15
51	7.1.2    DNS service capabilities .....	15
52	7.1.3    Managing the DNS service’s state.....	15
53	7.2    DNS service port management.....	16
54	7.2.1    UDP port .....	16
55	7.2.2    TCP listening port .....	17
56	7.3    DNS server representation .....	17
57	7.3.1    Relationship with DNS service.....	17
58	7.3.2    UDP port for DNS resolution session .....	17
59	7.3.3    TCP port for DNS resolution session.....	17
60	7.4    DNS service configuration .....	18
61	7.4.1    DNS service endpoint configuration .....	18
62	7.5    DNS service relationship with IP interfaces.....	18
63	7.5.1    Modeling the IP interface over which a DNS resolution session is established .....	18
64	7.5.2    Modeling the IP interfaces for the DNS service.....	19
65	8    Methods.....	19
66	8.1    CIM_ProtocolService.ListenOnPort( ).....	19
67	8.2    CIM_ProtocolService.ListenOnUDPPort( ).....	20
68	8.3    CIM_ProtocolService.RequestStateChange( ) .....	22
69	8.3.1    CIM_ProtocolService.RequestStateChange( ) conditional support.....	23
70	8.4    Profile conventions for operations .....	23
71	8.5    CIM_BindsTo .....	23
72	8.6    CIM_ElementCapabilities .....	24
73	8.7    CIM_ElementSettingData .....	24
74	8.7.1    CIM_ElementSettingData — ModifyInstance .....	24
75	8.8    CIM_HostedAccessPoint .....	25
76	8.9    CIM_HostedService .....	25
77	8.10    CIM_ProtocolService .....	25
78	8.10.1    CIM_ProtocolService — ModifyInstance .....	25
79	8.11    CIM_ProvidesEndpoint .....	26
80	8.12    CIM_ServiceAccessBySAP .....	26
81	8.13    CIM_IPProtocolEndpoint .....	26
82	8.14    CIM_TCPProtocolEndpoint.....	26
83	8.14.1    ModifyInstance.....	27
84	8.14.2    DeleteInstance.....	27
85	8.15    CIM_DNSServiceProtocolEndpoint .....	27

86	8.15.1	ModifyInstance.....	27
87	8.15.2	DeleteInstance.....	27
88	8.16	CIM_UDPProtocolEndpoint.....	28
89	8.16.1	ModifyInstance.....	28
90	8.16.2	DeleteInstance.....	28
91	8.17	CIM_DNSServiceCapabilities.....	28
92	8.18	CIM_DNSServiceSettingData.....	28
93	8.18.1	CIM_DNSServiceSettingData — ModifyInstance.....	29
94	9	Use cases.....	29
95	9.1	Profile Registration.....	29
96	9.2	Adding a UDP port for the DNS service.....	30
97	9.3	Determining the DNS service capabilities.....	31
98	9.3.1	Determine which IP address versions are supported.....	32
99	9.4	Configuring DNS service default settings.....	32
100	9.5	Modifying DNS service active settings.....	32
101	9.5.1	Enabling DNS service security policy.....	32
102	9.6	Viewing DNS service active settings.....	32
103	9.6.1	View the DNS service IP address.....	33
104	9.6.2	View the DNS resolution session policy.....	33
105	9.7	Stopping the DNS service on a specific UDP port.....	33
106	9.8	Disabling the DNS service.....	33
107	10	CIM Elements.....	34
108	10.1	CIM_BindsTo — CIM_IPProtocolEndpoint and CIM_TCPProtocolEndpoint.....	34
109	10.2	CIM_BindsTo — CIM_IPProtocolEndpoint and CIM_UDPProtocolEndpoint.....	35
110	10.3	CIM_BindsTo — CIM_TCPProtocolEndpoint and CIM_DNSServiceProtocolEndpoint.....	35
111	10.4	CIM_DNSServiceCapabilities.....	35
112	10.5	CIM_DNSServiceSettingData.....	37
113	10.6	CIM_DNSServiceProtocolEndpoint.....	38
114	10.7	CIM_ProtocolService.....	39
115	10.8	CIM_DNSServiceProtocolEndpointStats.....	40
116	10.9	CIM_ElementSettingData — CIM_ProtocolService and CIM_DNSServiceSettingData.....	40
117	10.10	CIM_ElementSettingData — CIM_DNSServiceProtocolEndpoint and CIM_DNSServiceSettingData.....	40
118	10.11	CIM_HostedAccessPoint — CIM_ComputerSystem and CIM_IPProtocolEndpoint.....	41
119	10.12	CIM_HostedAccessPoint — CIM_ComputerSystem and CIM_DNSServiceProtocolEndpoint.....	41
120	10.13	CIM_HostedService.....	42
121	10.14	CIM_IPProtocolEndpoint.....	42
122	10.15	CIM_RegisteredProfile.....	43
123	10.16	CIM_ElementCapabilities.....	43
124	10.17	CIM_ProvidesEndpoint.....	43
125	10.18	CIM_ServiceAccessBySAP - CIM_TCPProtocolEndpoint and CIM_ProtocolService.....	44
126	10.19	CIM_ServiceAccessBySAP - UDPProtocolEndpoint and CIM_ProtocolService.....	44
127	10.20	CIM_ElementConformsToProfile.....	44
128	10.21	CIM_TCPProtocolEndpoint.....	45
129	10.22	CIM_UDPProtocolEndpoint.....	45
130		ANNEX A (informative) Change log.....	46
131		Bibliography.....	47
132			
133			
134			

135 **Figures**

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

137 Figure 2 – Registered profile ..... 30

138 Figure 3 – UDP port configuration to specific interface ..... 31

139 **Tables**

140 Table 1 – Referenced profiles ..... 13

141 Table 2 – CIM\_ProtocolService.ListenOnPort( ) method: Return code values ..... 19

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

143 Table 4 – CIM\_ProtocolService.ListenOnUDPPort( ) method: Return code values ..... 21

144 Table 5 – CIM\_ProtocolService.ListenOnUDPPort( ) method: Parameters ..... 21

145 Table 6 – CIM\_ProtocolService.RequestStateChange( ) method: Return code values ..... 22

146 Table 7 – CIM\_ProtocolService.RequestStateChange( ) method: Parameters ..... 22

147 Table 8 – Operations: CIM\_BindsTo ..... 23

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

149 Table 10 – Operations: CIM\_ElementSettingData ..... 24

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

151 Table 12 – Operations: CIM\_HostedService ..... 25

152 Table 13 – Operations: CIM\_ProtocolService ..... 25

153 Table 14 – Operations: CIM\_ProvidesEndpoint ..... 26

154 Table 15 – Operations: CIM\_ServiceAccessBySAP ..... 26

155 Table 16 – Operations: CIM\_TCIPProtocolEndpoint ..... 26

156 Table 17 – Operations: CIM\_DNSServiceProtocolEndpoint ..... 27

157 Table 18 – Operations: CIM\_UDPProtocolEndpoint ..... 28

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

159 Table 20 – CIM Elements: DNS service management profile ..... 34

160 Table 21 – Class: CIM\_BindsTo — CIM\_IPProtocolEndpoint and CIM\_TCIPProtocolEndpoint ..... 35

161 Table 22 – Class: CIM\_BindsTo – CIM\_IPProtocolEndpoint and CIM\_UDPProtocolEndpoint ..... 35

162 Table 23 – Class: CIM\_BindsTo – CIM\_TCIPProtocolEndpoint and CIM\_DNSServiceProtocolEndpoint .. 35

163 Table 26 – Class: CIM\_DNSServiceSettingData ..... 37

164 Table 27 – Class: CIM\_DNSServiceProtocolEndpoint ..... 38

165 Table 28 – Class: CIM\_ProtocolService ..... 39

166 Table 29 – Class: CIM\_DNSServiceProtocolEndpointStats ..... 40

167 Table 30 – Class: CIM\_ElementSettingData — CIM\_ProtocolService and CIM\_DNSServiceSettingData 40

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

169 CIM\_DNSServiceSettingData ..... 40

170 Table 32 – Class: CIM\_HostedAccessPoint — CIM\_ComputerSystem and CIM\_IPProtocolEndpoint ..... 41

171 Table 33 – Class: CIM\_HostedAccessPoint — CIM\_ComputerSystem and

172 CIM\_DNSServiceProtocolEndpoint ..... 42

173 Table 34 – Class: CIM\_HostedService ..... 42

174 Table 35 – Class: CIM\_IPProtocolEndpoint ..... 42

175 Table 36 – Class: CIM\_RegisteredProfile ..... 43

176 Table 37 – Class: CIM\_ElementCapabilities ..... 43

177 Table 38 – Class: CIM\_ProvidesEndpoint ..... 44

178 Table 39 – Class: CIM\_ServiceAccessBySAP - CIM\_TCPProtocolEndpoint and CIM\_ProtocolService... 44  
179 Table 40 – Class: CIM\_ServiceAccessBySAP - UDPProtocolEndpoint and CIM\_ProtocolService ..... 44  
180 Table 41 – Class: CIM\_ElementConformsToProfile ..... 45  
181 Table 42 – Class: CIM\_TCPProtocolEndpoint..... 45  
182 Table 43 – Class: CIM\_UDPProtocolEndpoint ..... 45  
183

184

## Foreword

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

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

## 189 **Acknowledgments**

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

191 Editors:

- 192 • ZhongYu Gu – ZTE Corporation
- 193 • Ghazanfar Ali – ZTE Corporation

194 Contributors:

- 195 • Hemal Shah – Broadcom Corporation
- 196 • John Crandall – Brocade Communications System
- 197 • Alex Zhdankin – Cisco Systems
- 198 • Steve Neely – Cisco Systems
- 199 • Shishir Pardikar – Citrix
- 200 • John Parchem – Microsoft Corporation
- 201 • Lawrence Lamers – VMware
- 202 • Dr. Bhumip Khasnabish - ZTE Corporation

203

## Introduction

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

### 209 Document conventions

#### 210 Typographical conventions

211 The following typographical conventions are used in this document:

- 212 • Document titles are marked in *italics*.
- 213 • ABNF rules are in `monospaced font`.



214

# DNS Service Management Profile

## 215 1 Scope

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

## 220 2 Normative references

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

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

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

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

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

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

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

237 DMTF DSP1036 *IP Interface Profile 1.1.1*,  
238 [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)

239 DMTF DSP1038 *DNS Client Profile 1.0.3*,  
240 [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)

241 DMTF DSP1097, *Virtual Ethernet Switch Profile 1.1*,  
242 [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)

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

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

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

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

### 275 3 Terms and definitions

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

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

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

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

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

291 **3.1**

292 **conditional**

293 indicates requirements to be followed strictly to conform to the document when the specified conditions  
294 are met

295 **3.1**

296 **EDNS0**

297 Extension mechanisms for DNS

298 **3.2**

299 **DNSSEC**

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

303 **3.3**

304 **mandatory**

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

307 **3.4**

308 **optional**

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

310 **3.5**

311 **pending configuration**

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

314 **3.6**

315 **referencing profile**

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

318 **3.7**

319 **TSIG**

320 TSIG (Transaction SIGnature) is a computer networking protocol defined in RFC 2845

321 **3.8**

322 **unspecified**

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

## 324 4 Symbols and abbreviated terms

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

### 327 4.1

#### 328 IP

329 Internet Protocol

### 330 4.1

#### 331 DNS

332 Domain Name System

### 333 4.2

#### 334 DNSSEC

335 DNS Security Extensions

### 336 4.3

#### 337 TCP

338 Transmission Control Protocol

### 339 4.4

#### 340 TSIG

341 Transaction SIGNature

### 342 4.5

#### 343 UDP

344 User Datagram Protocol

## 345 5 Synopsis

346 **Profile name:** DNS Service Management Profile

347 **Version:** 1.0.0

348 **Organization:** DMTF

349 **CIM Schema version:** 2.46

350 **Central class:** CIM\_ProtocolService

351 **Scoping class:** CIM\_ComputerSystem

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

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

356

**Table 1 – Referenced profiles**

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

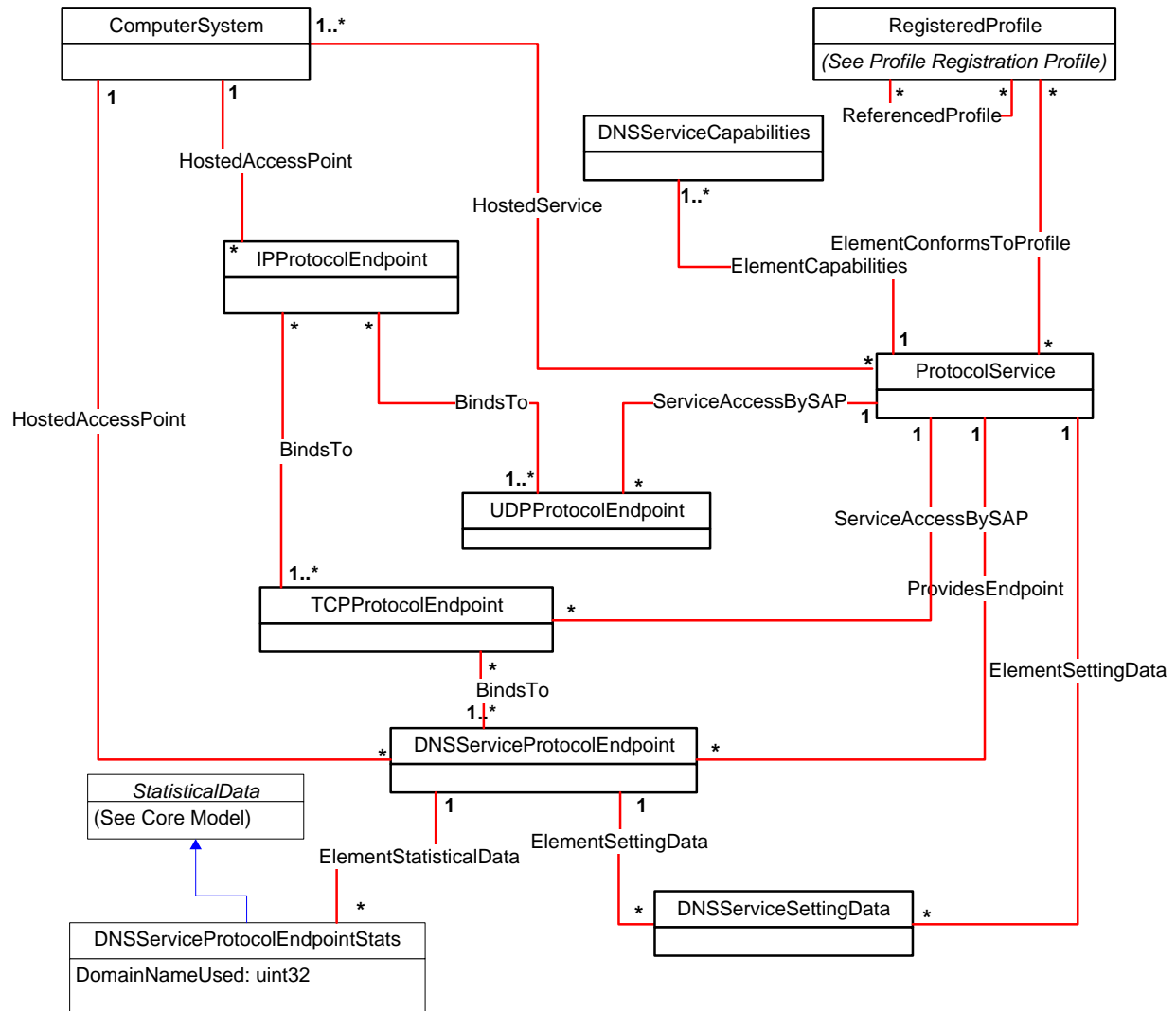
357 The central class for the *DNS Service Management Profile* shall be the CIM\_ProtocolService class. The  
 358 central instance of the *DNS Service Management Profile* shall be an instance of CIM\_ProtocolService.  
 359 The scoping class for the *DNS Service Management Profile* shall be CIM\_ComputerSystem.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

## 362 **6 Description**

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

### 366 **6.1 Class diagram**

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



369  
370

371 **Figure 1 – DNS Service Management Profile: Class diagram**

372 Functionality within the scope of this profile includes:

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

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

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

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

### 385 **6.1.1 Support for IP protocol versioning**

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

### 389 **6.1.2 Management of DNS security**

390 TSIG ([RFC2845](#)) is used to securely authenticate transactions and is configurable.

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

394 TSIG and DNSSEC can be used independently.

395 *[Open: DNS security management aspects.]*

### 396 **6.1.3 Representation of DNS service usage data**

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

398 *[Open: How do represent DNS statistics.]*

## 399 **7 Implementation**

### 400 **7.1 Representing a DNS service**

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

#### 402 **7.1.1 CIM\_ProtocolService.Protocol**

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

#### 404 **7.1.2 DNS service capabilities**

405 An instance of CIM\_DNSServiceCapabilities shall be associated with the CIM\_ProtocolService instance  
406 through an instance of CIM\_ElementCapabilities. This instance of CIM\_DNSServiceCapabilities shall  
407 represent the capabilities of the DNS service.

#### 408 **7.1.3 Managing the DNS service's state**

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

##### 411 **7.1.3.1 State management supported**

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

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

#### 416 **7.1.3.2 CIM\_ProtocolService.RequestStateChange( ) supported**

417 When the CIM\_DNSServiceCapabilities.RequestedStatesSupported property contains at least one value,  
418 the CIM\_ProtocolService.RequestStateChange( ) method shall be implemented and supported. The  
419 CIM\_ProtocolService.RequestStateChange( ) method shall not return a value of 1 (Unspecified).

#### 420 **7.1.3.3 RequestedState**

421 When the CIM\_ProtocolService.RequestStateChange( ) method is successfully invoked, the value of the  
422 RequestedState property shall be the value of the RequestedState parameter. If the method is not  
423 successfully invoked, the value of the RequestedState property is indeterminate.

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

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

#### 429 **7.1.3.4 EnabledState**

430 When the RequestedState parameter has a value of 2 (Enabled) or 3 (Disabled) and the  
431 CIM\_ProtocolService.RequestStateChange( ) method is completed successfully, the value of the  
432 EnabledState property shall equal the value of the CIM\_ProtocolService.RequestedState property.

433 If the method is not completed successfully, the value of the EnabledState property is indeterminate.

434 The EnabledState property shall have the value 2 (Enabled), 3 (Disabled), or 6 (Enabled but Offline).

#### 435 **7.1.3.5 Indicating state management support with CIM\_DNSServiceCapabilities**

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

## 440 **7.2 DNS service port management**

### 441 **7.2.1 UDP port**

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

#### 444 **7.2.1.1 CIM\_UDPProtocolEndpoint**

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

#### 447 **7.2.1.2 Relationship of UDP port to the DNS service**

448 An instance of CIM\_ServiceAccessBySAP shall associate the CIM\_ProtocolService instance with the  
449 CIM\_UDPProtocolEndpoint instance.



### 450 7.2.1.3 Managing UDP ports

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

## 455 7.2.2 TCP listening port

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

### 458 7.2.2.1 CIM\_TCPProtocolEndpoint

459 There shall be an instance of CIM\_TCPProtocolEndpoint in which the PortNumber property of the  
460 instance indicates the TCP port number on which the DNS service is listening.

### 461 7.2.2.2 Relationship of TCP port to the DNS service

462 An instance of CIM\_ServiceAccessBySAP shall associate the CIM\_ProtocolService instance with the  
463 CIM\_TCPProtocolEndpoint instance.

### 464 7.2.2.3 Managing TCP listening ports

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

## 469 7.3 DNS server representation

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

### 472 7.3.1 Relationship with DNS service

473 An instance of CIM\_ProvidesEndpoint shall associate the CIM\_ProtocolService with the  
474 CIM\_DNSServiceProtocolEndpoint.

### 475 7.3.2 UDP port for DNS resolution session

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

#### 478 7.3.2.1 CIM\_UDPProtocolEndpoint

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

### 481 7.3.3 TCP port for DNS resolution session

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

### 484 7.3.3.1 CIM\_TCPProtocolEndpoint

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

## 487 7.4 DNS service configuration

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

### 493 7.4.1 DNS service endpoint configuration

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

#### 498 7.4.1.1 Configuration that will be assigned

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

#### 505 7.4.1.2 Initial configuration of a DNS resolution session

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

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

## 515 7.5 DNS service relationship with IP interfaces

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

### 518 7.5.1 Modeling the IP interface over which a DNS resolution session is established

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

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

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

525 When the implementation models the specific interface over which a DNS resolution is active on TCP  
 526 port, there shall be an instance of the CIM\_BindsTo association where the value of the Antecedent  
 527 property shall be a reference to the CIM\_IPProtocolEndpoint instance and the value of the Dependent  
 528 property shall be a reference to the CIM\_UDPProtocolEndpoint instance.

529 **7.5.2 Modeling the IP interfaces for the DNS service**

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

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

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

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

543 **8 Methods**

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

546 **8.1 CIM\_ProtocolService.ListenOnPort( )**

547 The CIM\_ProtocolService.ListenOnPort( ) method shall be supported when the  
 548 ListeningPortManagement property of the associated instance of CIM\_DNSServiceCapabilities has a  
 549 value of TRUE. When the value of the ListeningPortManagement property is FALSE, the  
 550 CIM\_ProtocolService.ListenOnPort( ) method shall not be supported.

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

554 No standard messages are defined.

555

556 **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.

Value	Description
0x1000	Job started: REF returned to started CIM_ConcreteJob

557

**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 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

558 When the method is completed successfully, the implementation shall create an instance of  
 559 CIM\_TCPProtocolEndpoint. The value of the PortNumber property of the instance of  
 560 CIM\_TCPProtocolEndpoint shall be the value of the PortNumber parameter of the method invocation.  
 561 The implementation shall create an instance of CIM\_ServiceAccessBySAP that references the instance  
 562 of CIM\_TCPProtocolEndpoint and references the instance of CIM\_ProtocolService on which the method  
 563 was invoked.

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

- 565 • The implementation shall create an instance of CIM\_HostedAccessPoint that references the  
 566 newly created CIM\_TCPProtocolEndpoint instance and the instance of CIM\_ComputerSystem  
 567 with which the CIM\_ProtocolService instance is associated through an instance of  
 568 CIM\_HostedService (the scoping system).
- 569 • For each instance of CIM\_IPProtocolEndpoint that is associated through an instance of  
 570 CIM\_HostedAccessPoint with the CIM\_ComputerSystem instance with which the instance of  
 571 CIM\_ProtocolService on which this method was invoked is associated through an instance of  
 572 CIM\_HostedService, the implementation shall create an instance of the CIM\_BindsTo  
 573 association where the value of the Antecedent property shall be a reference to the  
 574 CIM\_IPProtocolEndpoint instance and the value of the Dependent property shall be a reference  
 575 to the CIM\_TCPProtocolEndpoint instance.

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

- 577 • The implementation shall create an instance of CIM\_HostedAccessPoint that references the  
 578 newly created CIM\_TCPProtocolEndpoint instance and the instance of CIM\_ComputerSystem  
 579 with which the CIM\_IPProtocolEndpoint instance is associated through an instance of  
 580 CIM\_HostedAccessPoint.
- 581 • The implementation shall create an instance of the CIM\_BindsTo association where the value of  
 582 the Antecedent property shall be a reference to the CIM\_IPProtocolEndpoint instance and the  
 583 value of the Dependent property shall be a reference to the CIM\_TCPProtocolEndpoint  
 584 instance.

## 585 8.2 CIM\_ProtocolService.ListenOnUDPPort( )

586 The CIM\_ProtocolService.ListenOnUDPPort( ) method shall be supported when the  
 587 ListeningPortManagement property of the associated instance of CIM\_DNSServiceCapabilities has a

588 value of TRUE. When the value of the ListeningPortManagement property is FALSE, the  
 589 CIM\_ProtocolService.ListenOnUDPPort ( ) method shall not be supported.

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

593 No standard messages are defined.

594 **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

595 **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

596 When the method is completed successfully, the implementation shall create an instance of  
 597 CIM\_UDPProtocolEndpoint. The value of the PortNumber property of the instance of  
 598 CIM\_UDPProtocolEndpoint shall be the value of the PortNumber parameter of the method invocation.  
 599 The implementation shall create an instance of CIM\_ServiceAccessBySAP that references the instance  
 600 of CIM\_UDPProtocolEndpoint and references the instance of CIM\_ProtocolService on which the method  
 601 was invoked.

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

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

- 614 The implementation shall perform the following actions when the IPEndpoint parameter is specified:
- 615 • The implementation shall create an instance of CIM\_HostedAccessPoint that references the
  - 616 newly created CIM\_UDPProtocolEndpoint instance and the instance of CIM\_ComputerSystem
  - 617 with which the CIM\_IPProtocolEndpoint instance is associated through an instance of
  - 618 CIM\_HostedAccessPoint.
  - 619 • The implementation shall create an instance of the CIM\_BindsTo association where the value of
  - 620 the Antecedent property shall be a reference to the CIM\_IPProtocolEndpoint instance and the
  - 621 value of the Dependent property shall be a reference to the CIM\_UDPProtocolEndpoint
  - 622 instance.

623 **8.3 CIM\_ProtocolService.RequestStateChange( )**

624 Invocation of the CIM\_ProtocolService.RequestStateChange( ) method will change the element’s state to

625 the value specified in the RequestedState parameter. The Enabled and Disabled values of the

626 RequestedState parameter correspond to enabling or disabling the functionality represented by the

627 instance of CIM\_ProtocolService. A value of 2 (Enabled) shall correspond to a request to enable the

628 functionality. A value of 3 (Disabled) shall correspond to a request to disable the functionality. A value of

629 11 (Reset) shall initiate a reset of the DNS service.

630 See 7.1.3.3 for information about the effect of this method on the RequestedState property.

631 The method shall be considered successful if the availability of the functionality upon completion of the

632 method corresponds to the desired availability indicated by the RequestedState parameter. An actual

633 change in state does not need to occur for the method to be considered successful. It is sufficient that the

634 resultant state be equal to the requested state. Upon successful completion of the method, the Return

635 Value shall be zero.

636 See 7.1.3.4 for information about the effect of this method on the EnabledState property.

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

638 No standard messages are defined.

639 Invoking the CIM\_ProtocolService.RequestStateChange( ) method multiple times could result in earlier

640 requests being overwritten or lost.

641 **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

642 **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

Qualifiers	Name	Type	Description/Values
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

643 **8.3.1 CIM\_ProtocolService.RequestStateChange( ) conditional support**

644 Support for the RequestStateChange( ) method is conditional on the indication of support for at least one  
 645 value for the RequestedState parameter as advertised through the RequestedStatesSupported property  
 646 of an associated instance of CIM\_EnabledLogicalElementCapabilities. When the  
 647 CIM\_EnabledLogicalElementCapabilities.RequestedStatesSupported property contains at least one  
 648 value, the CIM\_ProtocolService.RequestStateChange( ) method shall be implemented and supported.  
 649 The CIM\_ProtocolService.RequestStateChange( ) method shall not return a value of 1 (Unsupported).

650 **8.4 Profile conventions for operations**

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

653 The default list of operations is as follows:

- 654 • GetInstance
- 655 • EnumerateInstances
- 656 • EnumerateInstanceNames
- 657 • Associators
- 658 • AssociatorNames
- 659 • References
- 660 • ReferenceNames

661 **8.5 CIM\_BindsTo**

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

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

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

665 **8.6 CIM\_ElementCapabilities**

666 Table 9 lists implementation requirements for operations. If implemented, these operations shall be  
 667 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 9, all operations in  
 668 the default list in 8.2 shall be implemented as defined in [DSP0200](#).

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

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

670 **8.7 CIM\_ElementSettingData**

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

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

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

674 **8.7.1 CIM\_ElementSettingData — ModifyInstance**

675 When an instance of CIM\_ElementSettingData associates an instance of CIM\_DNSServiceSettingData  
 676 with an instance of CIM\_DNSServiceProtocolEndpoint, the following rules shall govern the behavior of  
 677 the ModifyInstance operation:

- 678 • The ModifyInstance operation shall not allow the IsDefault property to be modified.
- 679 • The ModifyInstance operation shall not allow the IsCurrent property to be modified.
- 680 • When the ModifyInstance operation is used to modify the IsNext property to a value of 1 (Is  
 681 Next), the ModifyInstance operation shall implement the following behavior:
  - 682 – The ModifyInstance operation shall find all other instances of CIM\_ElementSettingData  
 683 that associate a CIM\_DNSServiceSettingData instance with the  
 684 CIM\_DNSServiceProtocolEndpoint instance referenced by the target instance of  
 685 CIM\_ElementSettingData.
  - 686 – For each instance of CIM\_ElementSettingData found, the ModifyInstance operation shall  
 687 modify the value of its IsNext property to a value of 2 (Is Not Next).



688 **8.8 CIM\_HostedAccessPoint**

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

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

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

692 **8.9 CIM\_HostedService**

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

695 **Table 12 – Operations: CIM\_HostedService**

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

696 **8.10 CIM\_ProtocolService**

697 Table 13 lists implementation requirements for operations. If implemented, these operations shall be  
698 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 13, all operations  
699 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

701 **Table 13 – Operations: CIM\_ProtocolService**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.10.1.	None

702 **8.10.1 CIM\_ProtocolService — ModifyInstance**

703 When the ElementNameEditSupported property of the CIM\_DNSServiceCapabilities instance has a value  
704 of TRUE, the ModifyInstance operation shall allow the value of the ElementName property of the  
705 CIM\_ProtocolService instance to be modified. The ModifyInstance operation shall enforce the length  
706 restriction specified in the MaxElementNameLen property of the CIM\_DNSServiceCapabilities instance.  
707 When the ElementNameEditSupported property of the CIM\_DNSServiceCapabilities has a value of  
708 FALSE, the ModifyInstance operation shall not change the value of the ElementName property of the  
709 CIM\_ProtocolService instance.

710 **8.11 CIM\_ProvidesEndpoint**

711 Table 14 lists implementation requirements for operations. If implemented, these operations shall be  
 712 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 14, all operations  
 713 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

714 **Table 14 – Operations: CIM\_ProvidesEndpoint**

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

715 **8.12 CIM\_ServiceAccessBySAP**

716 Table 15 lists implementation requirements for operations. If implemented, these operations shall be  
 717 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 15, all operations  
 718 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

719 **Table 15 – Operations: CIM\_ServiceAccessBySAP**

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

720 **8.13 CIM\_IPProtocolEndpoint**

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

722 **8.14 CIM\_TCPProtocolEndpoint**

723 Table 16 lists implementation requirements for operations. If implemented, these operations shall be  
 724 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 16, all operations  
 725 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

726 **Table 16 – Operations: CIM\_TCPProtocolEndpoint**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.14.1.	None
DeleteInstance	Optional. See 8.14.2.	None

727 **8.14.1 ModifyInstance**

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

- 731 • NameFormat
- 732 • ProtocolIFType
- 733 • PortNumber

734 **8.14.2 DeleteInstance**

735 When the CIM\_ProtocolService.ListenOnPort( ) method is supported for the instance of  
 736 CIM\_ProtocolService with which the CIM\_TCPProtocolEndpoint is associated through an instance of  
 737 CIM\_ServiceAccessBySAP, the DeleteInstance operation shall be supported for the instance of  
 738 CIM\_TCPProtocolEndpoint. When the CIM\_ProtocolService.ListenOnPort( ) method is not supported, the  
 739 DeleteInstance operation shall not be supported.

740 When the DeleteInstance operation is successful for an instance of CIM\_TCPProtocolEndpoint, the DNS  
 741 service shall stop listening on the TCP/IP port indicated by the PortNumber property of the  
 742 CIM\_TCPProtocolEndpoint. The implementation shall also remove any association instances that  
 743 reference the CIM\_TCPProtocolEndpoint.

744 **8.15 CIM\_DNSServiceProtocolEndpoint**

745 Table 17 lists implementation requirements for operations. If implemented, these operations shall be  
 746 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 17, all operations  
 747 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

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

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.15.1.	None
DeleteInstance	Optional. See 8.15.2.	None

750 **8.15.1 ModifyInstance**

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

- 754 • NameFormat
- 755 • ProtocolIFType
- 756 • OtherTypeDescription

757 **8.15.2 DeleteInstance**

758 The DeleteInstance operation may be supported for instances of CIM\_DNSServiceProtocolEndpoint.  
 759 When the DeleteInstance operation is invoked against an instance, the corresponding DNS resolution  
 760 session shall be terminated prior to deleting the CIM\_DNSServiceProtocolEndpoint instance. The

761 implementation shall also remove any association instances that reference the  
762 CIM\_DNSServiceProtocolEndpoint.

## 763 8.16 CIM\_UDPProtocolEndpoint

764 Table 18 lists implementation requirements for operations. If implemented, these operations shall be  
765 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 18, all operations  
766 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

Operation	Requirement	Messages
ModifyInstance	Optional. See <b>Error! Reference source not found..</b>	None
DeleteInstance	Optional. See 8.16.2	

### 768 8.16.1 ModifyInstance

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

- 772 • NameFormat
- 773 • ProtocolFileType
- 774 • PortNumber

### 775 8.16.2 DeleteInstance

776 When the CIM\_ProtocolService.ListenOnUDPPort( ) method is supported for the instance of  
777 CIM\_ProtocolService with which the CIM\_UDPProtocolEndpoint is associated through an instance of  
778 CIM\_ServiceAccessBySAP, the DeleteInstance operation shall be supported for the instance of  
779 CIM\_UDPProtocolEndpoint. When the CIM\_ProtocolService.ListenOnUDPPort( ) method is not  
780 supported, the DeleteInstance operation should not be supported.

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

## 785 8.17 CIM\_DNSServiceCapabilities

786 All operations in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

## 788 8.18 CIM\_DNSServiceSettingData

789 Table 19 lists implementation requirements for operations. If implemented, these operations shall be  
790 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 19, all operations  
791 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

793

Table 19 – Operations: CIM\_DNSServiceSettingData

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.18.1.	None

794

### 8.18.1 CIM\_DNSServiceSettingData — ModifyInstance

795 When the CIM\_DNSServerSettingData instance is associated with the CIM\_ProtocolService instance  
 796 through an instance of CIM\_ElementSettingData and the value of the IsDefault property of the  
 797 CIM\_ElementSettingData instance has a value of 1 (Is Default), the ModifyInstance operation shall not be  
 798 supported.

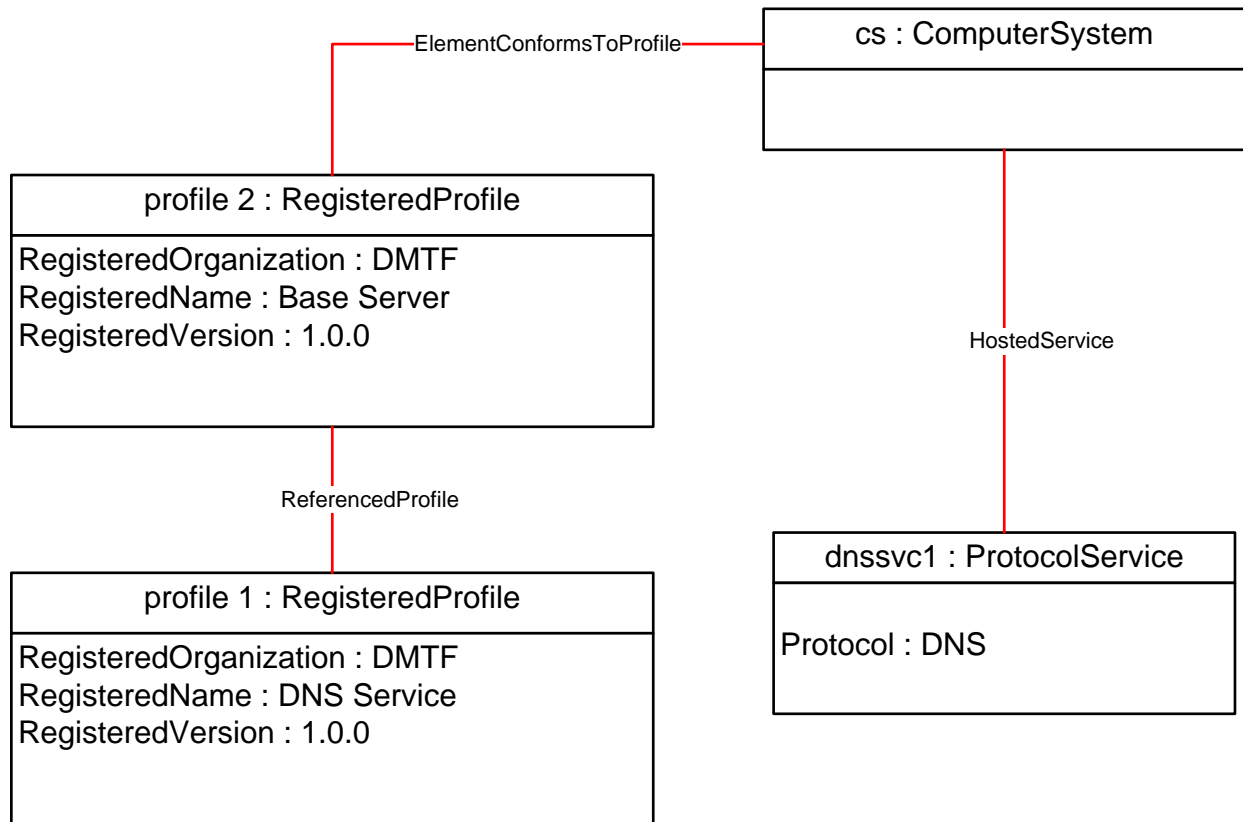
799 When the CIM\_DNSServerSettingData instance is not associated with an instance of CIM\_ProtocolService  
 800 through an instance of CIM\_ElementSettingData where the IsDefault property has a value of 1 (Is  
 801 Default), the ModifyInstance operation may be supported for the CIM\_DNSServerSettingData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

## 802 9 Use cases

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

### 804 9.1 Profile Registration

805 The object diagram in Figure 2 shows one possible method for advertising profile conformance. The  
 806 instances of CIM\_RegisteredProfile are used to identify the version of the *DNS Service Management*  
 807 *Profile* with which an instance of CIM\_ProtocolService is conformant. An instance of  
 808 CIM\_RegisteredProfile exists for each profile that is instrumented in the system. One instance of  
 809 CIM\_RegisteredProfile identifies the “DNS Service1.0.0”. The other instance identifies the “DNS Service  
 810 Management Profile”. The CIM\_ProtocolService instance is scoped to an instance of  
 811 CIM\_ComputerSystem.



812

813

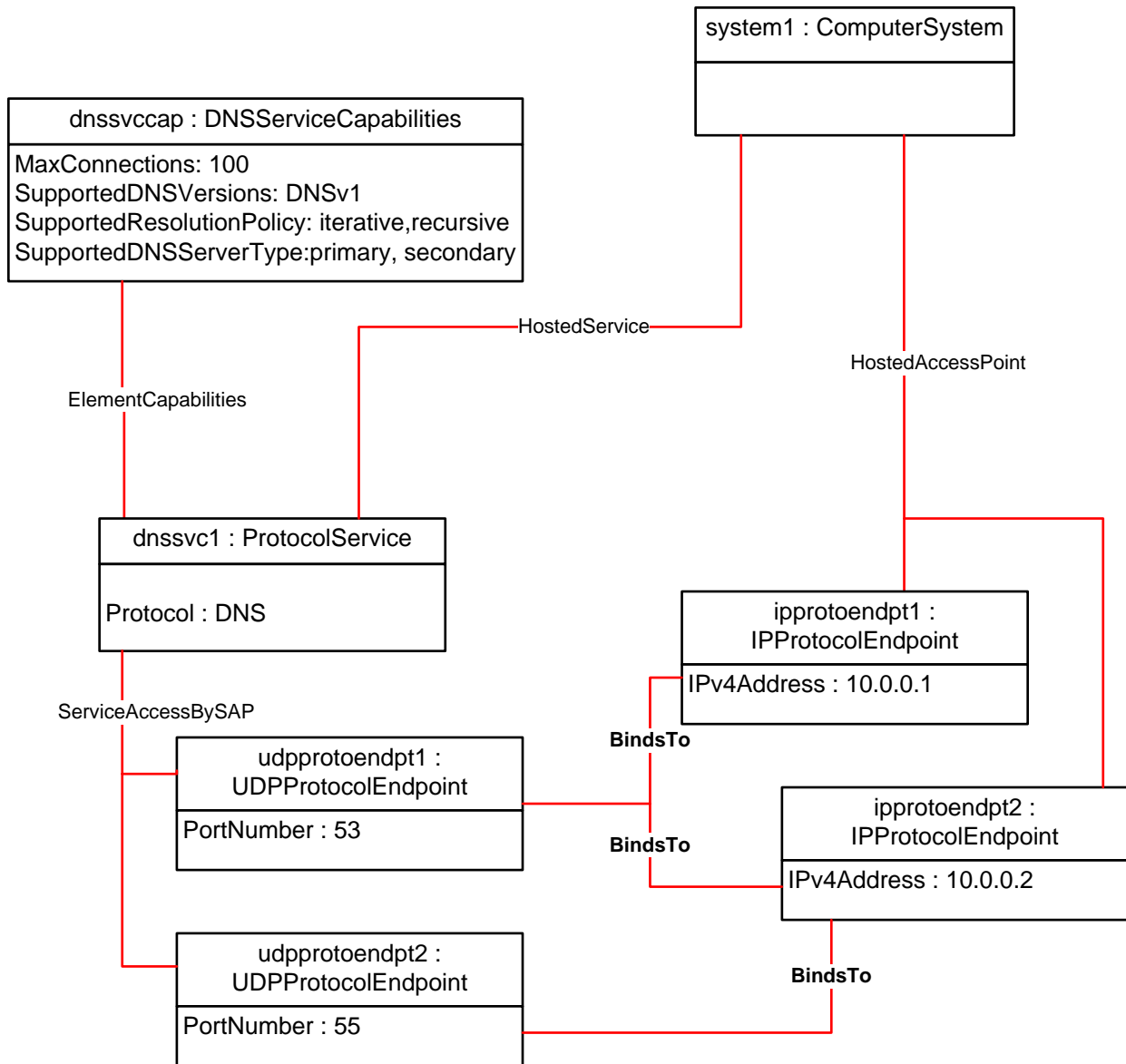
Figure 2 – Registered profile

814 **9.2 Adding a UDP port for the DNS service**

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

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

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



824

825

**Figure 3 – UDP port configuration to specific interface**

826

Figure 3 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 udprotoendpt2 is created and associated with ipprotoendpt2.

827

828

### 9.3 Determining the DNS service capabilities

829

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

- 831 1) Find the instance of CIM\_DNSServiceCapabilities associated with the CIM\_ProtocolService  
832 through an instance of CIM\_ElementCapabilities.
- 833 2) View the properties of the CIM\_DNSServiceCapabilities instance to see the supported  
834 functions.

### 835 9.3.1 Determine which IP address versions are supported

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

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

## 841 9.4 Configuring DNS service default settings

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

- 844 1) Find the instance of CIM\_ElementSettingData that associates an instance of  
845 CIM\_DNSServiceSettingData with the CIM\_ProtocolService, where the value of the IsDefault  
846 property is 1 (Is Default).
- 847 2) Modify the properties of the referenced CIM\_DNSServiceSettingData instance.

## 848 9.5 Modifying DNS service active settings

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

- 851 1) From the instance of CIM\_DNSServiceProtocolEndpoint, find the associated instance of the  
852 CIM\_DNSServiceSettingData through the CIM\_ElementSettingData association, where the  
853 value of the IsCurrent property is 1.
- 854 2) Modify the properties of the referenced CIM\_DNSServiceSettingData instance.

### 855 9.5.1 Enabling DNS service security policy

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

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

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

- 861 1) From the instance of CIM\_DNSServiceProtocolEndpoint, find the associated instance of the  
862 CIM\_DNSServiceSettingData through the CIM\_ElementSettingData association, where the  
863 value of the IsCurrent property is 1.
- 864 2) Set the values of TSIGEnable and/or DNSSecEnable properties of  
865 CIM\_DNSServiceSettingData instance equal to 1.

## 866 9.6 Viewing DNS service active settings

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



- 868 1) From the instance of CIM\_DNSServiceProtocolEndpoint, find the associated instance of the  
869 CIM\_DNSServiceSettingData through the CIM\_ElementSettingData association, where the  
870 value of the IsCurrent property is 1.
- 871 2) Instance of the CIM\_DNSServiceSettingData represents the active settings..

### 872 9.6.1 View the DNS service IP address

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

- 874 1) Find the instance of CIM\_DNSServiceProtocolEndpoint associated with either the  
875 CIM\_UDPProtocolEndpoint or CIM\_TCPProtocolEndpoint through an instance of CIM\_BindsTo.
- 876 2) Find the instance of CIM\_IPProtocolEndpoint associated with either the  
877 CIM\_UDPProtocolEndpoint or CIM\_TCPProtocolEndpoint through an instance of CIM\_BindsTo.
- 878 3) View the IPv4Address and IPv6Address properties of the CIM\_IPProtocolEndpoint instance to  
879 find the IP address of the DNS service.

### 880 9.6.2 View the DNS resolution session policy

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

- 882 1) From the instance of CIM\_DNSServiceProtocolEndpoint, find the associated instance of the  
883 CIM\_DNSServiceSettingData through the CIM\_ElementSettingData association, where the  
884 value of the IsCurrent property is 1.
- 885 2) View the value of DNSResolutionPolicy property of the CIM\_DNSServiceSettingData instance  
886 to find the DNS resolution session policy of the DNS service.

### 887 9.7 Stopping the DNS service on a specific UDP port

888 A management client can stop the DNS service from being accessible on a specific UDP port by invoking  
889 the intrinsic DeleteInstance operation against the instance of CIM\_UDPProtocolEndpoint that represents  
890 the UDP port.

891 Using the configuration shown in Figure 3 as an example, invoking the DeleteInstance operation against  
892 the instance udpprotoendpt2 would cause the DNS service to be no longer accessible on port 55.

### 893 9.8 Disabling the DNS service

894 If an implementation supports disabling the DNS service, a user can disable the DNS service by invoking  
895 the RequestStateChange( ) method on CIM\_ProtocolService instance with a value of Disabled for the  
896 RequestedState parameter.

897 **10 CIM Elements**

898 Table 20 shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be  
 899 implemented as described in Table 20. Clauses 7 (“Implementation”) and 8 (“Methods”) may impose  
 900 additional requirements on these elements.

901 **Table 20 – CIM Elements: DNS service management profile**

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

902 **10.1 CIM\_BindsTo — CIM\_IPProtocolEndpoint and CIM\_TCPProtocolEndpoint**

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

905 **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 *

906 **10.2 CIM\_BindsTo — CIM\_IPProtocolEndpoint and CIM\_UDPProtocolEndpoint**

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

909 **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 *

910 **10.3 CIM\_BindsTo — CIM\_TCPProtocolEndpoint and**  
911 **CIM\_DNSServiceProtocolEndpoint**

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

914 **Table 23 – Class: CIM\_BindsTo – CIM\_TCPProtocolEndpoint and**  
915 **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 *

916 **10.4 CIM\_DNSServiceCapabilities**

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

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.
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.

920 **10.5 CIM\_DNSServiceSettingData**

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

923 **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.
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.
EnabledDNSNotify	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.
EnabledDNSsec	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.

Elements	Requirement	Description
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.

924 **10.6 CIM\_DNSServiceProtocolEndpoint**

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

927

**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.

Elements	Requirement	Description
EnabledDNSsec	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.

928

929 **10.7 CIM\_ProtocolService**

930 CIM\_ProtocolService represents the DNS service. Table 27 provides information about the properties of  
 931 CIM\_ProtocolService.

932

**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 8.2.
ListenOnPort( )	Conditional	See 8.1.
ListenOnUDPPort( )	Conditional	See 8.2

933 **10.8 CIM\_DNSServiceProtocolEndpointStats**

934 The CIM\_DNSServiceProtocolEndpointStats represents the statistics of the DNS service that includes  
 935 total number of resolution requests to DNS server, the total number of requests resolved locally/by server  
 936 itself, the total number of requests resolved by support of other DNS server(s), usage frequency of  
 937 Domain Names, and so on.

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

939 **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.

940 **10.9 CIM\_ElementSettingData — CIM\_ProtocolService and**  
941 **CIM\_DNSServiceSettingData**

942 CIM\_ElementSettingData associates instances of CIM\_DNSServiceSettingData with the  
 943 CIM\_ProtocolService instance. Table 29 provides information about the properties of  
 944 CIM\_ElementSettingData.

945 **Table 29 – Class: CIM\_ElementSettingData — CIM\_ProtocolService and**  
946 **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 *

947 **10.10 CIM\_ElementSettingData — CIM\_DNSServiceProtocolEndpoint and**  
948 **CIM\_DNSServiceSettingData**

949 CIM\_ElementSettingData associates instances of CIM\_DNSServiceSettingData with the  
 950 CIM\_DNSServiceProtocolEndpoint instance. Table 30 provides information about the properties of  
 951 CIM\_ElementSettingData.

952 **Table 30 – Class: CIM\_ElementSettingData — CIM\_DNSServiceProtocolEndpoint and**  
953 **CIM\_DNSServiceSettingData**

Elements	Requirement	Description
ManagedElement	Mandatory	<b>Key:</b> This shall be a reference to the Central Instance. Cardinality *



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

954 **10.11 CIM\_HostedAccessPoint — CIM\_ComputerSystem and**  
 955 **CIM\_IPProtocolEndpoint**

956 CIM\_HostedAccessPoint associates an instance of CIM\_IPProtocolEndpoint with scoping  
 957 CIM\_ComputerSystem. Table 31 provides information about the properties of CIM\_HostedAccessPoint.

958 **Table 31 – Class: CIM\_HostedAccessPoint — CIM\_ComputerSystem 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 *

959 **10.12 CIM\_HostedAccessPoint — CIM\_ComputerSystem and**  
 960 **CIM\_DNSServiceProtocolEndpoint**

961 CIM\_HostedAccessPoint associates an instance of CIM\_DNSServiceProtocolEndpoint with scoping  
 962 CIM\_ComputerSystem. Table 32 provides information about the properties of CIM\_HostedAccessPoint.

963  
964**Table 32 – Class: CIM\_HostedAccessPoint — CIM\_ComputerSystem 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 *

965 **10.13 CIM\_HostedService**966 CIM\_HostedService relates the CIM\_ProtocolService instance to its scoping CIM\_ComputerSystem  
967 instance. Table 33 provides information about the properties of CIM\_HostedService.

968

**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 *

969 **10.14 CIM\_IPProtocolEndpoint**970 CIM\_IPProtocolEndpoint represents an IP interface that is associated with an Ethernet interface. Table 34  
971 provides information about the properties of CIM\_IPProtocolEndpoint.

972

**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.

973 **10.15 CIM\_RegisteredProfile**

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

979 **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".

980 **10.16 CIM\_ElementCapabilities**

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

983 **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

984 **10.17 CIM\_ProvidesEndpoint**

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

988

**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 *

989

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

990

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

994

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

995

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 *

996

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

997

998 CIM\_ServiceAccessBySAP is used to associate the instance of CIM\_ProtocolService with an instance of  
 999 CIM\_UDPProtocolEndpoint over which a DNS resolution session with the service can be established.  
 1000 Table 39 provides information about the properties of CIM\_ServiceAccessBySAP.

1001

**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 *

1002

### 10.20 CIM\_ElementConformsToProfile

1003

1004

1005

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

1006

**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

1007

**10.21 CIM\_TCIPProtocolEndpoint**

1008

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

1009

1010

**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

1011

**10.22 CIM\_UDPProtocolEndpoint**

1012

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

1013

1014

**Table 42 – Class: CIM\_UDPProtocolEndpoint**

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

**ANNEX A  
(informative)****Change log**

Version	Date	Description
1.0.0b	2015-06-19	DMTF Work in Progress

1015  
1016  
1017  
1018  
1019

1020

## Bibliography

1021

1022

1023