डोमेन नाम प्रणाली

डोमेन नाम प्रणाली ***(DNS)*** कंप्यूटर, सेवाओं, या किसी इंटरनेट या एक निजी नेटवर्क से जुड़े संसाधन के लिए एक क्रमिक नामकरण प्रणाली है। यह प्रतिभागी को दिए गये डोमेन नाम के साथ विभिन्न जानकारी एकत्रित करती है। सबसे महत्वपूर्ण बात यह है कि यह मनुष्यों के लिए अर्थपूर्ण डोमेन नामों को पूरी दुनिया में इन उपकरणों को पहचानने तथा संबोधित करने के प्रयोजन से नेटवर्किंग उपकरणों के साथ जुड़ी संख्यात्मक *(*बाईनरी*)* पहचान में बदल देती है। डोमेन नाम प्रणाली के बारे में अक्सर प्रयुक्त होने वाली कहावत यह है कि यह इंटरनेट के लिए "फ़ोन बुक" के रूप में मनुष्यों के अनुकूल कंप्यूटर होस्टनाम का आईपी एड्रेस के रूप में अनुवाद करती है। उदाहरण के लिए, *www.example.com* अनुवाद के बाद *208.77.188.166* हो जाता है।

डोमेन नाम प्रणाली इंटरनेट उपयोगकर्ताओं के समूह के लिए एक अर्थपूर्ण ढंग से डोमेन नाम निर्दिष्ट करना संभव बनाती है चाहे उपयोगकर्ता किसी भी स्थान पर हो। इस वजह से वर्ल्ड वाइड वेब *(WWW)* के हाईपरलिंक्स की और इंटरनेट संपर्क की जानकारी निरंतर तथा तटस्थ बनी रहती है चाहे वर्तमान इंटरनेट रूटिंग व्यवस्था में परिवर्तन हो जाए या उपयोगकर्ता मोबाइल उपकरण का प्रयोग करे. इंटरनेट डोमेन नामों को याद रखना *IP* एड्रेस याद रखने से ज्यादा आसान है जैसे *208.77.188.166 (IPv4)* या *2001:db8:1f70::999:de8:7648:6e8 (IPv6)*. लोग इस बात की परवाह किये बगैर अर्थपूर्ण यूआरएल और ई मेल पते बना कर इसका लाभ उठाते हैं कि मशीन *(*सर्वर*)* उन्हें कैसे ढूंढेगी.

डोमेन नाम प्रणाली डोमेन का नाम निर्धारित करने तथा उन नामों का *IP* पता आधिकारिक नाम सर्वर को निर्दिष्ट करके ढूँढने की जिम्मेवारी वितरित करती है। आधिकारिक नाम सर्वर अपने विशेष डोमेन के प्रति उत्तरदायी होते हैं और बदले में वे अपने उप-डोमेन के लिए अन्य आधिकारिक नाम सर्वर निर्धारित कर सकते हैं। इस तंत्र ने *DNS* को बांटने, त्रुटि सहने और लगातार सलाह तथा अपडेट से बचने के लिए एक केन्द्रीय रजिस्टर की आवश्यकता को नकारने योग्य बना दिया है।

सामान्यतः डोमेन नाम प्रणाली अन्य सूचनाओं का भी संग्रहण करती है जैसे मेल सर्वरों की सूची जो दिए गये इंटरनेट डोमेन के लिए ईमेल स्वीकार करती है। दुनिया भर में वितरित की जा सकने योग्य की-वर्ड आधारित पुनर्निर्धारण प्रणाली प्रदान करने की वजह से डोमेन नाम प्रणाली इंटरनेट की सुविधा का एक आवश्यक घटक है। दूसरे पहचानकर्ता जैसे कि *RFID* टैग, *UPC* कोड, ईमेल पतों और होस्ट नामों में अंतर्राष्ट्रीय वर्ण तथा विभिन्न प्रकार के दूसरे पहचानकर्ता संभावित रूप से *DNS* का प्रयोग

कर सकते हैं।[1] डोमेन नाम प्रणाली इस डाटाबेस सेवा की कार्यक्षमता के तकनीकी आधार भी परिभाषित करती है। इस प्रयोजन के लिए यह ***DNS*** प्रोटोकॉल को इंटरनेट प्रोटोकॉल सुइट ***(TCP/IP)*** के हिस्सों के रूप में ***DNS*** में प्रयुक्त होने वाली डाटा संरचनाओं तथा संचार एक्सचेंज का विस्तृत विवरण परिभाषित करती है। ***DNS*** प्रोटोकॉल को ***1980*** के दशक के आरम्भ में विकसित और परिभाषित किया गया तथा इंटरनेट इंजीनियरिंग टास्क फ़ोर्स द्वारा सार्वजनिक किया गया। *(*आगे देखें. इतिहास*).*

Internet Protocol Suite
Application Layer
BGP · DHCP · DNS · FTP · HTTP · IMAP · IRC · LDAP · MGCP · NNTP · NTP · POP · RIP · RPC · RTP · SIP · SMTP · SNMP · SOCKS · SSH · Telnet · TLS/SSL · XMPP · *(more)*
Transport Layer
TCP · UDP · DCCP · SCTP · RSVP · ECN · *(more)*
Internet Layer
IP (IPv4, IPv6) · ICMP · ICMPv6 · IGMP · IPsec · *(more)*
Link Layer
ARP/InARP · NDP · OSPF · Tunnels (L2TP) · PPP · Media Access Control (Ethernet, DSL, ISDN, FDDI) · *(more)*
द वा ब *(https://hi.wikipedia.org/w/index.php?title=%E0%A4%B8%E0%A4%BE%E0%A4%81%E0%A4%9A%E0%A4%BE:IPstack&action=edit)*

डोमेन नाम अगर साधारण शब्दो मे कहा जाए तो बो नाम जो किसी वेबसाइट के लिंक के साथ ऐड होता है जैसे विकिपीडिया एक नाम और जो ***.com*** है ये एक डोमेन है

इतिहास

नेटवर्क पर एक मशीन के संख्यात्मक पते के स्थान पर मनुष्य के पठन योग्य नामों का प्रचलन *TCP/IP* से भी पहले का है। यह प्रचलन *ARPAnet* युग का है। इसके बाद एक अलग प्रणाली का प्रयोग किया गया। *DNS* का अविष्कार *1983* में *TCP/IP* के शीघ्र बाद ही किया गया। पुराने सिस्टम में, नेटवर्क पर *SRI (*अब *SRI* इंटरनेशनल*)* पर स्थित प्रत्येक कंप्यूटर से एक कंप्यूटर *HOSTS.TXT* नामक फ़ाईल प्राप्त करता था।[2][3][4] *HOSTS.TXT* फ़ाईल में ढूंढे गये नामों का संख्यात्मक पता होता था। आधुनिक ऑपरेटिंग सिस्टम पर आज भी एक होस्ट फ़ाइल या तो डिफॉल्ट रूप में या सैटिंग के माध्यम से मौजूद होती है और उपयोगकर्ताओं को एक आईपी पता निर्धारित करने *(*उदाहरण *208.77.188.166)* और *DNS* की जाँच किये बिना एक होस्ट नाम के लिए अनुमति देती है। *(*उदाहरण *www.example.net).* होस्ट फ़ाइलों पर आधारित सिस्टम की कुछ मूलभूत सीमाएं हैं, क्योंकि यह एक स्पष्ट आवश्यकता है कि जब भी कंप्यूटर का *IP* पता बदले तो उससे संपर्क करने की कोशिश करने वाले कम्प्यूटरों को भी इसकी होस्ट फ़ाइल से अपडेट करने की आवश्यकता पड़ेगी.

नेटवर्किंग के विकास को और अधिक विश्वसनीय प्रणाली की आवश्यकता थी जो होस्ट में पते के बदलाव को केवल एक जगह पर रिकॉर्ड करे. दूसरे होस्ट एक सूचना प्रणाली के माध्यम से इस बदलाव के बारे में स्वयं जानकारी प्राप्त करेंगे जिससे सभी होस्ट नामों और उनसे सम्बंधित *IP* पतों तक पहुँच सुलभ हो जाएगी.

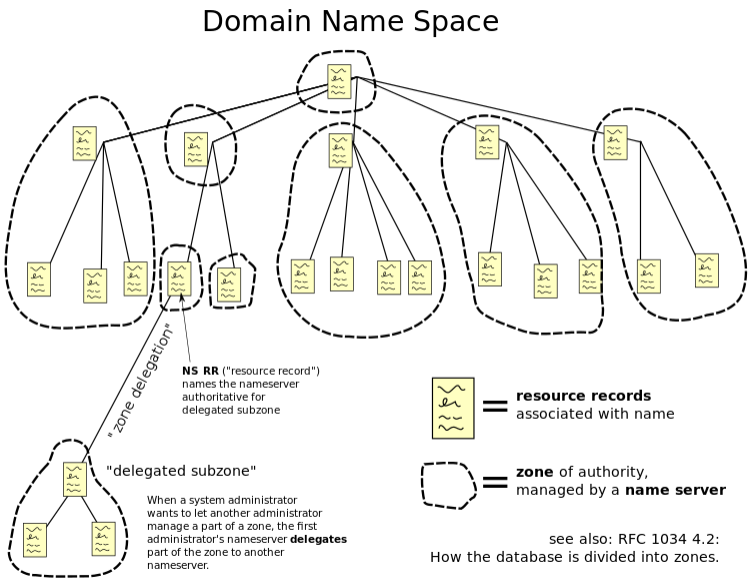
जॉन पोस्टेल के अनुरोध पर, पॉल मोकापेट्रिस ने *1983* में पहली डोमेन नाम प्रणाली का आविष्कार किया और इसे लागू करने का ढंग लिखा. मूल व्याख्या *RFC 882* और *RFC 883* में दिखाई गयी है जिन्हें नवम्बर *1987* में *RFC 1034* [5] और *RFC 1035* [6] से बदला गया। टिप्पणी के लिए कई अतिरिक्त अनुरोध मिलने के पर मूल *DNS* प्रोटोकॉल के विस्तार का प्रस्ताव किया गया है।

1984 में, चार बर्कले में पढ़ने वाले छात्रों डगलस टेरी, मार्क पेंटर, डेविड रिगल और सोंनिया झोउ ने पहला *UNIX* कार्यान्वन लिखा जिसे बाद में राल्फ कैंपबेल द्वारा पूरा किया गया। *1985* में, *DEC* के केविन डनलप ने पुनः *DNS* कार्यान्वयन लिखा और इसका नाम *BIND* रखा जिसका मतलब था बर्कले इंटरनेट नेम डोमेन. तब से माइक कैरेल, फिल एल्मक्विस्ट और पॉल विक्सी ने *BIND* को जिन्दा रखा है। *1990* के दशक के आरम्भ में *BIND* को विंडोज *NT* मंच पर स्थापित किया गया।

BIND को व्यापक रूप से, विशेषकर यूनिक्स सिस्टम पर वितरित किया गया और यह इंटरनेट पर प्रयोग होने वाला सबसे प्रभावी *DNS* सॉफ्टवेयर है।[7] अत्यधिक उपयोग और इसके मुक्त स्रोत कोड की जांच के परिणामस्वरूप, साथ ही साथ तेजी से और अधिक परिष्कृत हमले के तरीकों की वजह से *BIND* में कई सुरक्षा खामियां पाई गईं। इससे कई वैकल्पिक नेम सर्वर और रिसोल्वर प्रोग्रामों का विकास हुआ। *BIND* को स्क्रैच से संस्करण *9* में पुनः लिखा गया जिसका सुरक्षा रिकॉर्ड दूसरे आधुनिक इंटरनेट सॉफ्टवेयर से तुलना योग्य है।

संरचना

डोमेन नेम स्पेस



क्रमिक डोमेन नाम प्रणाली जो क्षेत्रों में संगठित है, प्रत्येक एक नाम सर्वर के द्वारा संचालित होती है।

डोमेन नेम स्पेस में डोमेन नामों का एक वृक्ष होता है। वृक्ष के प्रत्येक नोड या पत्ती में शून्य या अधिक *रिसोर्स रिकॉर्ड* होते हैं, जो डोमेन नाम के साथ संबद्ध जानकारी रखते है। वृक्ष रूट जोन में शुरुआत से *ज़ोन (क्षेत्रों)* में बंटा होता है। एक ***DNS*** ज़ोन में इससे जुड़े हुए नोड का एक संग्रह होता है जो आधिकारिक तौर पर एक *आधिकारिक नेम सर्वर* द्वारा भेजा जाता है। *(*ध्यान दें कि एक नेमसर्वर कई ज़ोन को होस्ट कर सकता है।*)*

किसी भी ज़ोन की प्रशासनिक जिम्मेदारी बांटी जा सकती है जिससे अतिरिक्त ज़ोन बनते हैं। आम तौर पर एक उप-डोमेन के रूप में पुराने स्पेस *(*स्थान*)* के एक हिस्से के लिए अधिकार दूसरे नेम सर्वर और प्रशासनिक इकाई को *सौंपे* ज़ा सकते हैं। पुराना ज़ोन नए ज़ोन के लिए अधिकृत नहीं रहता.

एक डोमेन नाम के हिस्से

एक डोमेन नाम आमतौर पर दो या अधिक भागों *(*तकनीकी *लेबल)* से बना होता है, जिन्हें पारंपरिक तौर पर डॉट से अलग किया जाता है जैसे ***example.com***।

- सबसे दायीं ओर का लेबल शीर्ष स्तर के डोमेन का पता बताता है *(*उदाहरण के लिए पता ***www.example.com*** में शीर्ष स्तर डोमेन ***com****)* है।
- बाईं ओर प्रत्येक लेबल एक सबडिविज़न या इसके बाद के डोमेन का उपडोमेन दर्शाता है। नोट: "उपडोमेन" आभासी निर्भरता व्यक्त करता है न कि पूर्ण निर्भरता. उदाहरण के लिए: ***example.com com*** डोमेन का उपडोमेन है और ***www.example.com*** डोमेन ***example.com*** का एक उपडोमेन है। सैद्धांतिक रूप में, यह सबडिविज़न *(*उपखंड*)* *127*

स्तरों तक जा सकता है। प्रत्येक लेबल में *63* ओक्टेट्स हो सकते हैं। पूरा डोमेन नाम *253* ओक्टेट्स की कुल लंबाई से अधिक नहीं हो सकता है।[8] व्यवहार में, कुछ डोमेन रजिस्ट्रियों की सीमा और भी कम हो सकती है।

- एक होस्ट नाम एक डोमेन नाम को दर्शाता है जिसके साथ एक या एक से अधिक आईपी पते *(*जैसे, *'www.example.com'* और *'example.com* डोमेन दोनों होस्ट नाम हैं, जबकि *'com'* डोमेन नहीं है*)* जुड़े होते हैं।

DNS सर्वर

डोमेन नाम प्रणाली को एक वितरित डाटाबेस प्रणाली द्वारा नियंत्रित किया जाता है है, जो क्लाइंट *(*ग्राहक*)* -सर्वर मॉडल का उपयोग करता है। इस डाटाबेस के नोड नाम सर्वर हैं। प्रत्येक डोमेन या उपडोमेन एक या एक से अधिक आधिकारिक *DNS* सर्वर हैं जो कि डोमेन और इसके अधीन किसी भी डोमेन के नाम सर्वर के बारे में जानकारी सार्वजनिक कर देते हैं। श्रंखल में शीर्ष पर रूट नेमसर्वर हैं : शीर्ष स्तर डोमेन नाम *(TLD)* देखते *(ढूंढते)* समय इनसे पूछताछ की जाती है।

DNS रिसोल्वर

इन्हें भी देखें: resolv.conf

DNS के क्लाइंट छोर को *DNS* रिसोल्वर कहा जाता है। यह मांगों को शुरू करने तथा उन्हें क्रमबद्ध करने के लिए जिम्मेवार है जिससे रिसोर्स *(*संसाधन*)* का पूर्ण हल *(*अनुवाद*)* किया जाता है जैसे डोमेन नाम का *IP* पते के रूप में अनुवाद.

एक *DNS* पूछताछ या तो एक गैर पुनरावर्ती *(*दोबारा न होने वाली*)* पूछताछ या पुनरावर्ती पूछताछ हो सकती है:

- एक *गैर पुनरावर्ती पूछताछ* वह है जिसमें *DNS* सर्वर एक डोमेन के लिए एक रिकॉर्ड प्रदान करता है जिसके लिए वह खुद प्राधिकृत है, या यह अन्य सर्वर से पूछे बिना एक आंशिक परिणाम प्रदान करता है।
- एक *पुनरावर्ती पूछताछ* वह है जिसके लिए डीएनएस सर्वर पूरी तरह से प्रश्न का जवाब आवश्यकतानुसार दूसरे नाम सर्वर से पूछ कर देगा *(*या एक त्रुटि दर्शायेगा*)*. *DNS* सर्वरों के लिए पुनरावर्ती प्रश्नों का समर्थन आवश्यक नहीं है।

रिसोल्वर या रिसोल्वर के स्थान पर पुनरावर्ती कर रहा अन्य *DNS* सर्वर क्वेरी हैडर में बिट्स के प्रयोग द्वारा पुनरावर्ती सेवा के उपयोग की बातचीत करता है।

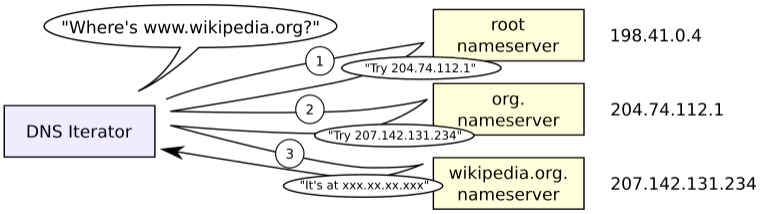
आम तौर पर यह रिसोल्विंग जरूरत की जानकारी प्राप्त करने के लिए कई नाम सर्वरों के बीच से गुजरती है। हालांकि, कुछ रिसोल्वर सरल ढंग से कार्य करते हैं और केवल एक नाम सर्वर के साथ ही संपर्क कर सकते हैं। ये सरल रिसोल्वर *(*"स्टब रिसोल्वर" कहा जाता है*)* जानकारी प्राप्त करने के लिए एक पुनरावर्ती नाम सर्वर पर निर्भर करते हैं।

प्रक्रिया

पता ढूँढने की तकनीक

एक डोमेन नाम में कई नाम घटक *(*जैसे, *ahost.ofasubnet.ofabiggernet.inadomain.example)* हो सकते हैं। व्यवहार में, पूरा होस्ट नाम अक्सर सिर्फ तीन घटकों से मिलकर बनता है: जैसे *ahost.inadomain.example,* और सबसे

अधिक बार *www. Inadomain.example.* पूछताछ प्रयोजनों के लिए, सॉफ्टवेयर सेगमेंट के दायें से बाईं ओर सेगमेंट की व्याख्या करता है। मार्ग में हर कदम पर, यह प्रोग्राम एक ***DNS*** सर्वर से अगले सर्वर का संकेत बताने का आग्रह करता है जिससे इसे परामर्श करना चाहिए।



*एक **DNS** रिकर्सर एड्रेस **www.wikipedia.org** का पता लगाने के लिए तीन नाम सर्वरों से सलाह लेता है।*

जैसे कि मूल रूप में परिकल्पित है, प्रक्रिया बहुत ही सरल है:

1. स्थानीय सिस्टम की पूर्व सैटिंग रूट सर्वरों में स्थित *रूट संकेतों* की फाइल के विश्वसनीय पतों से की जाती है जो कि स्रोत से स्थानीय एड्मिनिसट्रेटर द्वारा नवीनीकृत होने की साथ साथ अपडेट की जाती है।
2. रूट सर्वर से पहली पूछताछ अगले निचले स्तर के अधिकृत सर्वर को खोजने के लिए की जाती है *(*हमारे सरल होस्ट नेम के मामले में रूट सर्वर को, शीर्ष स्तर डोमेन ***example*** की विस्तृत जानकारी के साथ एक सर्वर के पते के लिए कहा जायेगा*).*
3. दूसरे स्तर के डोमेन की विस्तृत जानकारी के साथ एक ***DNS*** सर्वर के पते के लिए इस दूसरे सर्वर से पूछताछ होगी। हमारे उदाहरण में *(**inadomain.example**).*
4. पिछले चरण दोहराते हुए नीचे की ओर तब तक बढ़ना जब तक कि अंतिम चरण तक न पहुँच जायें जो कि अगले ***DNS*** सर्वर के पते के बजाए अंतिम ढूँढा गया हल *(*पता*)* होगा।

चित्र असली होस्ट ***www.wikipedia.org*** के लिए इस प्रक्रिया को दिखाता है।

इस सरल रूप तंत्र में एक मुश्किल है: यह मूल सर्वर पर भारी बोझ डालता है, क्योंकि एक पते की प्रत्येक खोज के साथ प्रत्येक सर्वर के साथ पूछताछ की शुरुआत हो जाती है। एक सिस्टम की सम्पूर्ण कार्यप्रणाली में महत्वपूर्ण होने के नाते, प्रत्येक दिन में अरबों प्रश्नों का बोझ एक दुर्गम अड़चन पैदा करेगा। व्यवहार में इस समस्या पर काबू पाने के लिए कैशिंग का प्रयोग होता है और वास्तव में, रूट नेम सर्वर को कुल यातायात के बहुत कम हिस्से का सामना करना पड़ता है।

सर्कुलर निर्भरता और ग्लू रिकॉर्ड

डेलीगेशन में नाम सर्वर *IP* पते से सूचीबद्ध होने की बजाए नाम से प्रदर्शित होते हैं। इसका अर्थ यह है कि एक रिसोल्विंग नाम सर्वर को एक निर्दिष्ट किये गये सर्वर के *IP* पते को ढूँढने के लिए एक अन्य *DNS* अनुरोध अवश्य भेजना चाहिए। चूंकि इस वजह से सर्कुलर निर्भरता की स्थिति हो सकती है, यदि एक डोमेन के तहत एक नेम सर्वर को निर्दिष्ट किया जाये जो इसके अधीन है, तो यह आवश्यक है कि ऐसे मामले में नेम सर्वर डेलीगेशन को अगले नेम सर्वर का *IP* पता भी अवश्य प्रदान करे. यह रिकॉर्ड एक *ग्लू रिकॉर्ड* कहलाता है।

उदाहरण के लिए मानिए कि उप-डोमेन ***en.wikipedia.org*** के और भी उप-डोमेन हैं *(*जैसे ***something.en.wikipedia.org****)* और यह कि इनका आधिकारिक नाम सर्वर ***ns1.something.en.wikipedia.org*** है। एक कंप्यूटर जो ***ns1.something.en.wikipedia.org*** को ढूँढने का प्रयास कर रहा है, उसे पहले ***something.en.wikipedia.org*** को ढूँढने का प्रयास करना होगा। चूंकि ***Ns1*** भी ***something.en.wikipedia.org*** उपडोमेन के अंतर्गत है, इसलिए ***ns1.something.en.wikipedia.org*** को ढूँढने के लिए ***something.en.wikipedia.org*** को ढूंढना पड़ेगा जो कि निश्चित तौर पर एक सर्कुलर निर्भरता है, जैसे कि ऊपर बताया गया है। इस निर्भरता को ***en.wikipedia.org*** के नेम सर्वर के ग्लू रिकॉर्ड द्वारा तोडा जाता है जो कि अनुरोधकर्ता को सीधे ही ***ns1.something.en.wikipedia.org*** का *IP* पता प्रदान करता है जो ***ns1.somethingen.wikipedia.org*** को ***bootstrap*** प्रक्रिया द्वारा ढूँढने की प्रक्रिया के लिए सक्षम बनाता है।

कैशिंग और जीवन समय

क्योंकि डीएनएस की तरह एक प्रणाली द्वारा उत्पन्न अनुरोधों की मात्रा विशाल है, इसके चलते, डिजाइनरों ने प्रत्येक डीएनएस सर्वर पर बोझ को कम करने के लिए एक तंत्र प्रदान करने का निश्चय किया। आज तक, *DNS* रेसोल्यूशन प्रक्रिया एक सफल जवाब के बाद एक दी गयी अवधि तक *कैशिंग (*अर्थात् एक *DNS* पूछताछ के परिणामों की स्थानीय रिकॉर्डिंग और इससे जुड़ी सलाह*)* के लिए अनुमति देती है। कितने समय तक एक *DNS* रिसोल्वर एक *DNS* प्रतिक्रिया को कैशे में रखता है *(*अर्थात कितने समय तक एक *DNS* प्रतिक्रिया *मान्य* रहती है*)*, यह एक मूल्य से निर्धारित होता है जिससे प्रतिक्रिया का जीवन समय *(*टीटीएल*)* कहा जाता है। टीटीएल *DNS* प्रतिक्रिया बाहर सौंपने वाले सर्वर के एडमिनिस्ट्रेटर द्वारा तय किया जाता है। अवधि की वैधता कुछ सेकंड से दिन या कुछ सप्ताहों तक भी हो सकती है।

कैशिंग समय

इस वितरित और कैशिंग संरचना के एक उल्लेखनीय परिणाम के रूप में, *DNS* रिकॉर्ड में परिवर्तन हमेशा तुरंत और दुनिया भर में प्रभावी नहीं होते. इसे एक उदाहरण के माध्यम से अच्छी तरह से स्पष्ट किया ज़ा सकता है : यदि किसी एडमिनिसट्रेटर ने होस्ट *www.wikipedia.org* के लिए *6* घंटे का टीटीएल सेट किया है और इसके बाद *IP* पता जिसमें *12:01pm* पर *www.wikipedia.org* को ढूँढा जाना है, एडमिनिस्ट्रेटर को यह सोचना चाहिए कि वह व्यक्ति जिसने *12:00* बजे दोपहर में पुराने आईपी पते से एक प्रतिक्रिया को कैश्ड किया होगा, वह *6:00pm* तक *DNS* सर्वर से संपर्क नहीं कर सकेगा। इस उदाहरण में *12:01pm* और *6:00pm* के बीच की अवधि को *कैशिंग समय* कहा जाता है, जो कि एक ऐसे समय के रूप में सर्वश्रेष्ठ ढंग से परिभाषित है जो तब शुरू होता है जब आप *DNS* रिकॉर्ड में बदलाव करते हैं तथा तब समाप्त होता है जब टीटीएल द्वारा निर्दिष्ट समय की अधिकतम सीमा समाप्त हो जाती है। डीएनएस में परिवर्तन करते समय इसे एक महत्वपूर्ण

तार्किक सोच के रूप में देखा जाता है : *यह आवश्यक नहीं कि प्रत्येक वा देख रहा हो जो आप देख रहे हैं।* *RFC 1912* टीटीएल स्थापित करने के बुनियादी नियमों को बताने में मदद करता है।

शब्द "प्रसार" पर ध्यान दें हालांकि यह इस संदर्भ में बहुत व्यापक रूप से प्रयुक्त होता है, अच्छी तरह से कैशिंग के प्रभाव का वर्णन नहीं करता है। विशेष रूप से, इसका अर्थ है कि *[1]* जब आप एक *DNS* परिवर्तन करते हैं, यह किसी तरह अन्य *DNS* सर्वरों में फ़ैल जाता है *(*बजाए इसके कि आपकी ज़रुरत के समय अन्य *DNS* सर्वर आप के साथ जांच करें*)* और *[2]* कि आपका कैश्ड किये गये रिकॉर्ड के समय की मात्रा पर कोई नियंत्रण नहीं है *(*आप *NS* रिकॉर्ड और आपके डोमेन नाम का उपयोग करने वाले दूसरे आधिकारिक *DNS* सर्वर के अलावा अपने डोमेन में सभी *DNS* रिकॉर्ड के लिए टीटीएल मूल्य नियंत्रित करते हैं*).*

कुछ रिसोल्वर टीटीएल मूल्यों को भी पार कर जाते हैं क्योंकि प्रोटोकॉल कैशिंग के लिए *68* साल तक या बिल्कुल कैशिंग नहीं, का समर्थन करता है। नकारात्मक कैशिंग *(*रिकॉर्ड का अस्तित्व में न होना*)* एक ज़ोन के अधिकृत नाम सर्वर द्वारा निर्धारित की जाती है जिसमें स्टार्ट ऑफ़ ऑथोरिटी *(SOA)* रिकॉर्ड का अवश्य उल्लेख किया जाना चाहिए जब अनुरोध के जवाब में कोई भी डाटा रिपोर्टिंग के लिए उपलब्ध नहीं है। *SOA* रिकार्ड के न्यूनतम क्षेत्र और खुद *SOA* के टीटीएल का प्रयोग नकारात्मक जवाब के लिए टीटीएल स्थापित करने में होता है। *RFC 2308*

जब आप एक *DNS* में परिवर्तन करते हैं तो बहुत से लोग गलत तरीके से रहस्यमय *48* घंटे या *72* घंटे को प्रचार समय मानते हैं। यदि कोई किसी डोमेन का *DNS* रिकॉर्ड बदल दे या एक डोमेन के अधिकृत *DNS* सर्वरों के होस्ट नामों का *IP* पता *(*यदि कोई हो तो*)* बदल दे सभी *DNS* सर्वरों को नयी सूचना का प्रयोग करने से पहले लम्बा समय लग सकता है। यह इसलिए हैं क्योंकि वो रिकॉर्ड ज़ोन पैरेन्ट *DNS* सर्वरों *(*उदाहरण के लिए। *Com* डीएनएस सर्वर यदि आपका डोमेन *example.com* है*),* द्वारा नियंत्रित किये जा रहे हैं जो आम तौर पर *48* घंटे के लिए उन रिकॉर्ड को कैश *(*संग्रहित*)* करते हैं। हालांकि, ये *DNS* परिवर्तन किसी भी *DNS* सर्वर के लिए तत्काल उपलब्ध होंगे जिसने उन्हें कैश्ड नहीं किया होगा। और आपके डोमेन पर *NS* रिकॉर्ड और अधिकृत *DNS* सर्वर नाम के अलावा कोई भी *DNS* परिवर्तन लगभग तुरंत होगा, यदि आप ऐसे करना चाहते हैं। *(TTL* को एक या दो बार समय से आगे कर के और तब तक प्रतीक्षा कर के जब तक की परिवर्तन से पहले पुराना *TTL* ख़त्म नहीं हो जाता*).*

रिवर्स लुकअप *(*खोज*)*

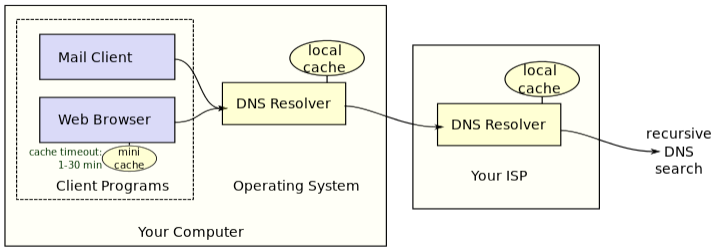
"रिवर्स लुकअप" शब्द दिए गये *IP* पते से जुड़े नाम को खोजने के लिए *DNS* पूछताछ को संदर्भित करता है।

DNS विशेष डोमेन में *IP* पते को *PTR* रिकॉर्ड के रूप में संग्रहित करता है। *IPv4* के लिए, डोमेन *in-addr.arpa* है। *IPv6* के लिए, रिवर्स लुकअप डोमेन *ip6.arpa* है।

रिवर्स लुकअप की प्रक्रिया करते समय, *DNS* क्लाइंट पते को *DNS* में प्रयुक्त प्रारूप में बदल देता है, तथा इसके बाद हमेशा की तरह डेलिगेशन श्रंखला का पालन करता है। उदाहरण के लिए, *IPv4* पता *'208.80.152.2'*, *2.152.80.208.in-addr.arpa* में बदल जाता है। *DNS* रिसोल्वर की पूछताछ रूट सर्वरों से होती है जो *208.in-addr.arpa* ज़ोन के लिए *ARIN* सर्वर की ओर संकेत करता है। वहाँ से *152.80.208.in* के लिए विकीमिडिया सर्वर नियुक्त किये गये हैं और *PTR*

विकीमिडिया नेम सर्वर से *2.152.80.208.in-addr.arpa* के लिए पूछताछ करके लुकअप *(*खोज*)* पूर्ण करता है, जिसके परिणामस्वरूप एक एक आधिकारिक परिणाम मिलता है।

क्लाइंट/ग्राहक लुकअप



DNS निर्णय क्रम.

उपयोगकर्ता आम तौर पर सीधे एक *DNS* रिसोल्वर के साथ संपर्क नहीं करते. इसके बजाय *DNS* रेसोल्यूशन एप्लीकेशन प्रोग्राम में पारदर्शी ढंग से जगह बनाता है जैसे कि वेब ब्राउज़र, ई मेल ग्राहक और अन्य इंटरनेट एप्लीकेशन/प्रोग्राम. प्रोग्राम एप्लीकेशन के एक ऐसे अनुरोध जिसके लिए डोमेन नेम लुकअप की आवश्यकता है, ऐसे प्रोग्राम रेसोल्यूशन अनुरोध को स्थानीय ऑपरेटिंग सिस्टम में *DNS* रिसोल्वर को भेजते हैं जो बदले मैं आवश्यक संचार नियंत्रित करता है।

DNS रिसोल्वर में लगभग हमेशा ही हाल में प्रयुक्त किये गये लुकअप का कैश/संग्रह *(*ऊपर देखें*)* होता है। यदि कैश अनुरोध का उत्तर दे पाए तो रिसोल्वर कैश का उत्तर अनुरोध करने वाले प्रोग्राम को देगा। यदि कैश में उत्तर न हो तो रिसोल्वर एक या अधिक निर्दिष्ट *DNS* सर्वरों को अनुरोध भेज देगा। ज्यादातर घरेलू उपयोगकर्ताओं के मामले में, इंटरनेट सेवा प्रदाता जिससे मशीन संपर्क स्थापित करती है, आम तौर पर इस *DNS* सर्वर की आपूर्ति करता है: ऐसे उपयोगकर्ता का सर्वर पता मैन्युअल रूप से कॉन्फ़िगर होगा या *DHCP* को निर्धारित करने की अनुमति देगा, लेकिन जहां सिस्टम एडमिनिस्ट्रेटर द्वारा अपने *DNS* सर्वर प्रयोग करने के लिए सिस्टम कॉन्फ़िगर किये गये हैं, उनका *DNS* रिसोल्वर संगठन के अलग रखे गये नेम सर्वर की ओर संकेत करता है। किसी भी परिस्थिति में, पूछताछ किये जाने वाला नेम सर्वर ऊपर दी गयी प्रक्रिया का पालन करेगा जब तक कि सफलतापूर्वक परिणाम का पता न लग जाए अथवा न लगा सके। तब यह अपना परिणाम *DNS* रिसोल्वर को यह मानते हुए भेज देता है कि इसने उत्तर ढूंढ लिया है, रिसोल्वर उत्तर को भविष्य के लिए कैश्ड *(*संग्रहित*)* कर लेता है और उत्तर को उस सॉफ्टवेर के पास वापिस भेज देता है जिसने शुरुआत में प्रश्न पूछा था।

टूटे हुए रिसोल्वर

जब रिसोल्वर *DNS* प्रोटोकॉल के नियमों का उल्लंघन करता है तो एक अतिरिक्त स्तर की जटिलता उभरती है। बड़ी संख्या में इंटरनेट सेवा प्रदाताओं *(ISPs)* ने अपने *DNS* सर्वर नियमों का उल्लंघन करने के लिए यह मानते हुए सेट कर रखे हैं कि

उन्हें पूर्ण नियमबद्ध रिसोल्वर के बजाए एक कम महंगा हार्डवेयर चलाने की अनुमति मिली हुई है।[9]

जटिलता के अंतिम स्तर के रूप में, कुछ एप्लीकेशन *(*जैसे कि वेब ब्राउज़र*)* के भी अपने ***DNS*** कैश होते है, ताकि वे स्वयं ***DNS*** रिसोल्वर पुस्तकालय/लाईब्रेरी का प्रयोग कम करें। इसमें अतिरिक्त मुश्किल तब और जुड़ जाती है जब ***DNS*** मुद्दों की डिबगिंग *(*गल्तियाँ सुधारना*)* होती है, क्योंकि यह डाटा की ताजगी का पता लगाना और/या यह बताना कि डाटा किस कैश से आ रहा है, कठिन बना देता है। ये कैश आम तौर पर बहुत कम कैशिंग समय - एक मिनट के आदेश पर - काम करते हैं। इंटरनेट एक्सप्लोरर एक उल्लेखनीय अपवाद प्रस्तुत करता है : ***recent*** के अनुसार संस्करण आधे घंटे तक ***DNS*** रिकॉर्ड को कैश करता है।[10]

दूसरे प्रोग्राम/एप्लीकेशन

ऊपर उल्लेख की गयी प्रणाली कुछ सरल परिदृश्य प्रदान करती है। डोमेन नाम प्रणाली के कुछ अन्य कार्य हैं :

- यह आवश्यक नहीं कि होस्टनेम और *IP* पता एक दूसरे से वन टू वन आधार पर *(*बिल्कुल*)* मेल खाएं. कई होस्टनेम एक *IP* पते से संपर्क कर सकते हैं जो आभासी होस्टिंग के द्वारा जुड़ा हुआ है, इससे एक मशीन कई वेब साइटों पर संयुक्त उप से काम करने की अनुमति देती है। वैकल्पिक रूप से एक होस्टनेम कई *IP* पतों के साथ संपर्क कर सकता है : इससे गल्ती सहने और लोड वितरण में सहायता मिलती है तथा यह बिना त्रुटी के एक साइट को भौतिक स्थान पर ले जाने की अनुमति देता है।
- नाम को *IP* पते में अनुवादित करने के अलावा ***DNS*** के कई उपयोग हैं। उदाहरण के लिए, मेल ट्रांसफर एजेंट ***DNS*** का प्रयोग यह पता लगाने के लिए करते हैं कि एक विशेष पते के लिए ई-मेल कहाँ भेजी जाए. डोमेन से मेल एक्सचेंजर तक का रास्ता ***MX*** रिकॉर्ड द्वारा उपब्ध कराया जाता है जो नाम के *IP* पते तक पहुँचने के लिए गल्ती सहने और लोड वितरित करने वाली एक और परत को स्थान देता है।
- ई-मेल ब्लैकलिस्ट *(*काली सूची*)*: ***DNS*** प्रणाली का कुशल ढंग से उपयोग काली सूची में डाले गये ई-मेल होस्ट के *IP* पते को संग्रहित करने तथा वितरित करने के लिए किया जाता है। इसकी सामान्य विधि यह है कि उस होस्ट के *IP* पते को एक उच्च स्तर डोमेन के उप-डोमेन में दाल दिया जाता है और विभिन्न रिकार्डों को सकारात्मक या नकारात्मक दिखने के लिए उस नाम का प्रयोग किया जाता है। ***blacklist.com*** का एक निम्न काल्पनिक उदाहरण है,
  - *102.3.4.5* को काली सूची में डाला गया *=> 5.4.3.102.**blacklist.com*** बनाता है और *127.0.0.1* पर डालता है।
  - *102.3.4.6* नहीं है *=> 6.4.3.102.**blacklist.com*** नहीं मिलता, या डिफ़ॉल्ट *127.0.0.2*
  - इसके पश्चात् ई-मेल सर्वर ***blacklist.com*** से ***DNS*** प्रणाली के माध्यम से यह पता करने के लिए पूछताछ कर सकते हैं कि उनसे जुड़ने वाला विशिष्ट होस्ट काली सूची में है कि नहीं। आज इस तरह की कई काली सूचियाँ, या तो मुफ्त या सदस्यता के आधार पर, ईमेल एड्मिनिस्ट्रेट्रों और स्पैम विरोधी सॉफ्टवेयर के प्रयोग के लिए मुख्य रूप से उपलब्ध हैं।
- सॉफ्टवेयर अपडेट: कई एंटी वायरस और वाणिज्यिक सॉफ्टवेयर अब नवीनतम सॉफ्टवेयर के अपडेट के लिए ***DNS*** प्रणाली का प्रयोग करते हैं जिससे क्लाइंट सर्वर को हर समय अपडेट सर्वर से जुड़ने की ज़रुरत नहीं पड़ती. इस तरह की एप्लीकेशन के लिए ***DNS*** रिकॉर्ड का कैश समय आम तौर पर अपेक्षाकृत कम होता है।

- अपने रिकॉर्ड टाइप बनाने के बजाए प्रेषक नीति की रूपरेखा और डोमेन की (चाबी) जो कि *TXT* रिकॉर्ड है, को दूसरे *DNS* रिकार्ड टाइप लाभ उठाने के लिए डिज़ाइन किया गया था।
- कंप्यूटर के फेल होने की स्थिति में लचीलापन प्रदान करने के लिए, प्रत्येक डोमेन को कवर करने के लिए आम तौर पर कई *DNS* सर्वर उपलब्ध कराए जाते हैं और शीर्ष स्तर पर तेरह अत्यधिक शक्तिशाली रूट सर्वर हैं, तथा एनीकास्ट के जरिए उनमें से कई की अतिरिक्त "प्रतिलिपियाँ" दुनिया भर में वितरित की गयी हैं।
- डाइनेमिक *DNS* (*DDNS* भी कहा जाता है) ग्राहकों को गतिशीलता के कारण *DNS* बदलने की स्थिति में, अपना *IP* पता अपडेट करने की क्षमता प्रदान करते हैं।

प्रोटोकॉल विवरण

DNS मुख्यत: पोर्ट संख्या *53*[11] पर यूज़र डाटाग्राम प्रोटोकॉल (*UDP*) का प्रयोग अनुरोध का उत्तर देने के लिए करता है। *DNS* पूछताछ में ग्राहक द्वारा पूछे गये एक *UDP* प्रश्न का जवाब सर्वर द्वारा एक *UDP* परिणाम द्वारा दिया जाता है। ट्रांसमिशन कंट्रोल प्रोटोकॉल (*TCP*) का प्रयोग तब किया जाता है जब उत्तर के रूप में डाटा का आकार *512* बाइट्स से अधिक है या फिर ज़ोन स्थानान्तरण जैसे कार्यों में किया जाता है। कुछ ऑपरेटिंग सिस्टम जैसे कि *HP-UX* में कुछ ऐसे रिसोल्वर लागू किये गये हैं जो कि सभी तरह की पूछताछ में *TCP* का तब भी प्रयोग करते हैं, जब इसके लिए केवल *UDP* ही पर्याप्त है।

डीएनएस रिसोर्स/संसाधन रिकॉर्ड

अधिक जानकारी: *[[List of DNS record types]]*

एक *रिसोर्स रिकॉर्ड* (*RR*) डोमेन नाम प्रणाली में मूलभूत डाटा तत्व है। प्रत्येक रिकॉर्ड में एक प्रकार (*A*, *MX*, आदि), एक समय समाप्ति सीमा, एक वर्ग और कुछ विशेष प्रकार के डाटा होते हैं। एक ही प्रकार के रिसोर्स रिकॉर्ड एक *रिसोर्स रिकॉर्ड सेट* को परिभाषित करते हैं। एक सेट में रिसोर्स रिकॉर्ड का आदेश अपरिभाषित है, जो रिसोल्वर द्वारा एप्लीकेशन को भेजा जाता है, लेकिन अक्सर सर्वर लोड संतुलन प्राप्त करने के लिए राउण्ड रॉबिन आदेश लागू करते हैं। बहरहाल *DNSSEC*, एक वैधानिक क्रम में पूरे रिसोर्स रिकॉर्ड पर काम करता है।

एक *IP* नेटवर्क पर भेजे जाने वाले सभी रिकॉर्ड *RFC 1035* और नीचे दिखाए गये आम प्रारूप का प्रयोग करते हैं।

RR (संसाधन रिकॉर्ड) क्षेत्र		
क्षेत्र	**विवरण**	**लंबाई ओक्टेट्स**
नाम	उस नोड का नाम जिससे यह रिकॉर्ड संबंधित है।	वेरिएबल
प्रकार	आर आर का प्रकार. उदाहरण के लिए, *MX* का प्रकार *15* है।	*2*
वर्ग	वर्ग कोड.	*2*
TTL	सेकंड में अहस्ताक्षरित समय जो *RR* द्वारा मान्य रहता है, अधिकतम *2147483647* है।	*4*
RDLENGTH	*RDATA* क्षेत्र की लंबाई.	*2*
RDATA	अतिरिक्त *RR* - विशेष डाटा.	वेरिएबल

NAME/नाम एक वृक्ष के नोड का पूरी तरह से योग्य डोमेन नाम है। वायर पर, लेबल संपीड़न का प्रयोग करके नाम को छोटा किया जा सकता है जहाँ वर्तमान डोमेन नाम का छोर पैकेट में पहले से ही उल्लेख किये गये मौजूदा डोमेन नाम के छोर से बदले जा सकते हैं।

type/टाइप रिकॉर्ड का प्रकार है। यह डाटा का स्वरूप बताता है और अपने उद्देश्य का संकेत देता है। उदाहरण के लिए, *A* रिकॉर्ड का प्रयोग डोमेन नाम को एक *IPv4* पते पर अनुवाद करने के लिए किया जाता है, *NS* रिकॉर्ड यह सूची प्रदर्शित करता है कि एक *DNS* ज़ोन में कौन से नाम सर्वर लुकअप का जवाब दे सकते हैं और *MX* रिकॉर्ड उस मेल सर्वर को निर्दिष्ट करता है जो एक ई-मेल पते में निर्दिष्ट डोमेन के लिए मेल को नियंत्रित करता है। *(DNS* रिकॉर्ड प्रकारों की सूची भी देखें*).*

RDATA एक टाइप विशेष डाटा है जैसे एड्रेस/पता रिकॉर्ड के लिए *IP* पते के रूप में या प्राथमिकता और एमएक्स रिकॉर्ड के लिए होस्ट नाम के रूप में. ज्ञात रिकॉर्ड टाइप *RDATA* क्षेत्र में लेबल संपीड़न का प्रयोग कर सकते हैं, लेकिन "अज्ञात" प्रकारों को नाघिं करना चाहिए *(RFC 3597).*

इंटरनेट होस्टनेम, सर्वर या *IP* पते में में शामिल आम *DNS* रिकॉर्ड के लिए, रिकॉर्ड के वर्ग को *IN (इंटरनेट)* के लिए सेट किया जाता है। इसके अतिरिक्त, *CH (चाओस)* और *HS (हेसिओड)* वर्ग भी मौजूद हैं। प्रत्येक वर्ग *DNS* ज़ोन के संभावित विभिन्न देलिगेशनों के साथ एक पूरी तरह से स्वतंत्र वृक्ष है।

जोन फाइल में परिभाषित रिसोर्स रिकॉर्ड के अतिरिक्त डोमेन नाम प्रणाली अनुरोधों के कई प्रकार भी परिभाषित करती है, जो केवल अन्य *DNS* नोड के साथ संचार में प्रयुक्त होते हैं *(तार पर),* जैसे ज़ोन स्थानान्तरण *(AXFR/IXFR)* या *EDNS (OPT)* के लिए।

वाइल्डकार्ड *DNS* रिकॉर्ड

डोमेन नाम प्रणाली *वाइल्डकार्ड डोमेन नाम* का समर्थन करती है जो ऐसे नाम हैं जो *एस्टरिस्क लेबल, '*'*, के साथ शुरू होते हैं जैसे, ****.example***। [5][12] वाइल्डकार्ड डोमेन नाम से सम्बंधित ***DNS*** रिकॉर्ड एक ***DNS*** ज़ोन में रिसोर्स नाम उत्पन्न करने के लिए, सभी लेबलों में से पूछे गये नाम से मिलते जुलते नाम घटा कर, जिसमें निर्दिष्ट वंशावली भी शामिल है, नियम निर्धारित करता है। उदाहरण के लिए, ***DNS*** ज़ोन ***x.example*** में, निम्नलिखित सेटअप बताता है कि ***x.example*** के सभी उपडोमेन ***(***जिनमें उपडोमेन के उपडोमेन भी शामिल हैं***)*** मेल एक्सचेंजर ***a.x.example*** का उपयोग करते हैं। मेल एक्सचेंजर को निर्दिष्ट करने के लिए ***a.x.example*** के लिए रिकॉर्ड की जरूरत है। चूंकि यह परिणाम वाइल्डकार्ड के मिलान से डोमेन नेम तथा इसे उपडोमेन छोड़ कर प्राप्त हुआ है, ***a.x.example*** के सभी उपडोमेन वाइल्डकार्ड परिणामों में अवश्य परिभाषित होने चाहिएं.

```
X.EXAMPLE. MX 10 A.X.EXAMPLE.
*.X.EXAMPLE. MX 10 A.X.EXAMPLE.
*.A.X.EXAMPLE. MX 10 A.X.EXAMPLE.
A.X.EXAMPLE. MX 10 A.X.EXAMPLE.
A.X.EXAMPLE. AAAA 2001:db8::1
```

वाइल्डकार्ड रिकॉर्ड की भूमिका ***RFC 4592*** में परिष्कृत की गयी थी क्योंकि ***RFC 1034*** में मूल परिभाषा अधूरी थी और जिसके कारण इसको लागू करने वाले सही ढंग से व्याख्या नहीं कर प् रहे थे।[12]

प्रोटोकॉल एक्सटेंशन

मूल ***DNS*** प्रोटोकॉल में नई सुविधाओं के विस्तार के लिए सीमित प्रावधान थे। ***1999*** में पॉल विक्सी ने ***RFC 2671*** में एक विस्तार प्रणाली सार्वजनिक की जिसे ***DNS*** के लिए विस्तार तंत्र***(EDNS)*** कहा जाता है जिसमें बिना खर्च बढ़ाए प्रयुक्त न होने वाले वैकल्पिक प्रोटोकॉल तत्व थे। ऐसा बनावटी रिसोर्स रिकॉर्ड ऑप्ट के माध्यम से किया गया जो केवल प्रोटोकॉल के तार प्रसारण में मौजूद था, किन्तु किसी भी ज़ोन फाइलों में नहीं था। प्रारंभिक विस्तारों का भी सुझाव दिया गया ***(EDNS0)***, जैसे ***UDP*** डाटाग्राम में ***DNS*** संदेश के आकार में वृद्धि.

डाइनेमिक ज़ोन अपडेट

एक आधिकारिक ***DNS*** सर्वर पर ज़ोन डाटा बेस में रखे रिकार्डों को डाइनेमिक ***DNS*** अपडेट के अपडेट ***DNS opcode*** की सहायता से जोड़ या हटा कर, डाइनेमिक ढंग से अपडेट करता है। यह सुविधा ***RFC 2136*** में वर्णित है। इस सुविधा से ***DNS*** में नेटवर्क के ग्राहकों को रजिस्टर किया जाता है जब वे बूट करते हैं या नेटवर्क पर उपलब्ध होते हैं। चूंकि बूटिंग करने वाले ग्राहक को हर बार ***DHCP*** सर्वर द्वारा एक अलग ***IP*** पत सौंपा जा सकता है, इस तरह के ग्राहकों के लिए स्थिर ***DNS*** उपलब्ध कराना संभव नहीं है।

अंतर्राष्ट्रीयकृत डोमेन नाम

यद्यपि तकनीकी रूप से डोमेन नाम में प्रयुक्त किये जाने वाले वर्णों पर कोई प्रतिबंध नहीं है और उसमें गैर-*ASCII* वर्ण भी शामिल हो सकते हैं, पर होस्ट नाम के लिए यह लागू नहीं होता। [13] होस्ट नामों को ज्यादातर लोग ई-मेल और वेब ब्राउज़िंग के लिए देखते और प्रयोग करते हैं। होस्ट नाम *ASCII* वर्णों के एक छोटे सबसेट तक ही सीमित हैं जिन्हें **LDH** के रूप में जाना जाता है, **L** *A-Z* तक छोटे और बड़े अक्षर, **D** संख्या *0-9* तक, **H** हाइफन, और *LDH* लेबलों को अलग करने के लिए डॉट; विस्तृत जानकारी के लिए *RFC 3696* खंड *2* देखें. इससे कई स्थानीय भाषाओं के नामों तथा शब्दों के प्रदर्शन पर अंकुश लग गया। *ICANN* ने पूनीकोड आधारित *IDNA* प्रणाली को मंज़ूरी दे दी है, जो इस विषय पर काम करते हुए यूनिकोड वाक्यों को वैध वर्ण समूह में बदलता है। कुछ रजिस्ट्रियों ने भी *IDNA* को स्वीकार कर लिया है।

सुरक्षा मुद्दे

शुरुआत में *DNS* को बनाते समय सुरक्षा को ध्यान में नहीं रखा गया था।

कमियों का एक स्वरूप *DNS* कैश दूषित होना है, जो *DNS* सर्वर को यह विश्वास दिलाता है कि प्रामाणिक जानकारी प्राप्त हो गयी है जबकि वास्तविकता में ऐसा नहीं होता।

पारंपरिक रूप से *DNS* परिणाम क्रिप्टोग्राफी द्वारा हस्ताक्षरित नहीं होते जिससे हमले की कई संभावनाएं बढ़ जाती हैं; डोमेन नाम प्रणाली सुरक्षा विस्तार *(DNSSEC)*, *DNS* को क्रिप्टोग्राफी द्वारा हस्ताक्षरित परिणाम देने के लिए परिवर्तित कर देता है। ज़ोन स्थानान्तरण जानकारी का समर्थन करने कई विस्तार भी उप्लभ हैं।

यहां तक कि एन्क्रिप्शन द्वारा भी, एक *DNS* सर्वर एक वायरस के साथ समझौता कर *(*या कहें कि असंतुष्ट कर्मचारियों द्वारा*)* जो कि सर्वर के *IP* पते को एक लम्बे *TTL* के साथ एक गलत पते पर निर्देशित करेंगे। यह इंटरनेट उपयोगकर्ताओं के लाखों लोगों पर दूरगामी संभावित प्रभाव डाल सकते हैं, अगर व्यस्त *DNS* सर्वर खराब *IP* डाटा को कैश *(*संग्रहित*)* कर लेता है। इसे ख़त्म करने के लिए सभी प्रभावित *DNS* कैश को मैन्युअल ढंग से साफ़ करना पड़ेगा जो लम्बे *TTL (68* साल तक*)* तक हो सकता है।

कुछ डोमेन नाम दूसरे, मिलते जुलते डोमेन नामों के द्वारा धोखा दे सकते हैं। उदाहरण के लिए, *"paypal.com"* और *"paypa1.com"* अलग नाम हैं, फिर भी उपयोगकर्ता अंतर बताने में असमर्थ हो सकता है जब उपयोगकर्ता का टाइप फेस *(*फ़ॉन्ट*)* स्पष्ट रूप से वर्ण *l* और अंक *1* में अंतर नहीं दिखाता. यह समस्या उन सिस्टमों में अधिक गंभीर है जो अन्तराष्ट्रीयकृत डोमेन नामों का समर्थन करते हैं। चूंकि कई अक्षर हैं जो *ISO 10646* के दृष्टिकोण से अलग हैं, कुछ खास कंप्यूटर स्क्रीन पर समान दिखाई देते हैं। इस कमजोरी का फायदा अक्सर फ़िशिंग में उठाया जाता है।

कुछ फॉरवर्ड कनफर्म्ड रिवर्स *DNS* तकनीकों द्वारा भी *DNS* परिणाम मान्य करने में सहायता मिलती है।

.

डोमेन नाम पंजीकरण

एक डोमेन नाम का प्रयोग करने का अधिकार डोमेन नाम रजिस्ट्रार द्वारा प्रदान किया जाता है जिन्हें नाम और संख्याओं के लिए इन्टरनेट निगम *(ICANN)* द्वारा मान्यता दी जाती है, एक ऐसा संगठन जो इंटरनेट के नाम और संख्या प्रणाली की देखरेख के हेतु प्रतिबद्ध है। *ICANN* के अतिरिक्त, प्रत्येक शीर्ष स्तर डोमेन *(TLD)*, जो रजिस्ट्री को ऑपरेट करता है, की देखरेख और तकनीकी सर्विस एक प्रशासनिक संगठन द्वारा की जाती है। एक रजिस्ट्री अपने अधीन *TLD* में पंजीकृत नाम के डाटाबेस के रखरखाव के लिए जिम्मेदार है। रजिस्ट्री एक डोमेन नाम रजिस्ट्रार, जो इसी *TLD* में नाम आवंटित करने के लिए अधिकृत है, से पंजीकरण के लिए जानकारी प्राप्त करती है और एक विशेष सेवा, *whois* प्रोटोकॉल का उपयोग करके जानकारी को प्रकाशित/सार्वजनिक करती है।

रजिस्ट्री और रजिस्ट्रार आम तौर पर एक उपयोगकर्ता के लिए एक डोमेन नाम प्रदान करने और एक नाम सर्वर के डिफ़ॉल्ट सेट उपलब्ध कराने की सेवा के लिए एक वार्षिक शुल्क लेते हैं। अक्सर इस लेनदेन को डोमेन नाम की बिक्री या लीज़ कहा जाता है और रजिस्ट्रेन्ट को "मालिक" भी कहा जा सकता है, परन्तु वास्तव में इस लेनदेन के साथ ऐसा कोई कानूनी संबंध जुड़ा हुआ नहीं है, केवल डोमेन नाम को प्रयोग करने का एकमात्र विशिष्ट अधिकार मिलता है। अधिक सही ढंग से, अधिकृत उपयोगकर्ता "पंजीकृत" या "डोमेन धारकों" के रूप में जाने जाते हैं।

ICANN TLD रजिस्ट्रियों और डोमेन नाम रजिस्ट्रारों की एक पूरी सूची दुनिया में प्रकाशित करती है। कोई भी कई डोमेन रजिस्ट्रियों द्वारा रखे गये *WHOIS* डाटाबेस में देख कर डोमेन नाम के धारक के बारे में जानकारी प्राप्त कर सकता है।

240 से अधिक देशों के कोड शीर्ष स्तर डोमेन *(ccTLDs)* के लिए, डोमेन रजिस्ट्रियां आधिकारिक *WHOIS* जानकारी रखती हैं। *(*धारक, नाम सर्वर, समाप्ति तिथियाँ, आदि*).* उदाहरण के लिए,*DENIC,* जर्मनी *NIC* आधिकारिक *WHOIS* को। *DE* डोमेन नाम में रखती है। *2001* के बाद से ज्यादातर *gTLD* रजिस्ट्रियों *(.Org,.BIZ,.INFO)* ने इस तथाकथित "मोटी" रजिस्ट्री दृष्टिकोण को अपनाया है, अर्थात् बजाय पंजीयकों के आधिकारिक *WHOIS* को केंद्रीय रजिस्ट्रियों में रखा है।

COM और *NET* डोमेन नामों के लिए, एक "पतली" रजिस्ट्री प्रयुक्त होती है: डोमेन रजिस्ट्री *(*जैसे *VeriSign)* एक बुनियादी *WHOIS (*रजिस्ट्रार और नाम सर्वर, आदि*)* रखती है। कोई भी विस्तृत *WHOIS (*धारक, नाम सर्वर, समाप्ति तिथि आदि*)* पंजीयक से प्राप्त कर सकता है।

कुछ डोमेन नाम रजिस्ट्रियां, जिन्हें अक्सर *नेटवर्क सूचना केन्द्र (NIC)* कहा जाता है, भी उपयोगकर्ता के लिए पंजीयकों के रूप में कार्य करती हैं। प्रमुख सामान्य शीर्ष स्तर डोमेन रजिस्ट्रियां जैसे *COM, NET, ORG, INFO* डोमेन और अन्य के लिए, एक रजिस्ट्री-रजिस्ट्रार डोमेन नामक मॉडल का प्रयोग करता है जिसमें सैकड़ों डोमेन नाम रजिस्ट्रार शामिल हैं *(ICANN (https://web.archive.org/web/20080622034433/http://www.icann.org/registrars/accredited-list.html)* या *VeriSign (https://web.archive.org/web/20090628122132/http://www.verisign.com/information-services/naming-services/com-net-registry/page_002166.html)* पर सूची देखें*).* प्रबंधन की इस विधि में, रजिस्ट्री केवल डोमेन नाम डाटाबेस और रजिस्ट्रार के साथ सम्बन्ध स्थापित करती है। *पंजीकृत* या *रजिस्ट्रेंट (*डोमेन नाम उपयोगकर्ता*)* रजिस्ट्रार के ग्राहक हैं, जो कुछ मामलों में पुनर्विक्रेताओं की अतिरिक्त परतों के माध्यम से भी हो सकते हैं।

एक डोमेन नाम दर्ज करने और नये नाम पर अधिकार बनाए रखने की प्रक्रिया में, रजिस्ट्रार एक डोमेन के साथ जुड़ी सूचना के कई प्रमुख हिस्सों का उपयोग करता है:

- *प्रशासनिक संपर्क.* एक रजिस्ट्रेंट आम तौर पर डोमेन नाम के प्रबंधन के लिए एक प्रशासनिक संपर्क निर्दिष्ट करता है। प्रशासनिक संपर्क का आमतौर पर एक डोमेन पर उच्चतम स्तर का नियंत्रण होता है। प्रशासनिक संपर्कों को सौंपे गये कई प्रबंधन कार्यों में व्यावसायिक जानकारी के प्रबंधन से जुड़ी सभी सूचनाएं हो सकती हैं जैसे कि नाम के रूप में रिकॉर्ड, डाक पता और अधिकृत पंजीकृत *(*रजिस्ट्रेंट*)* की संपर्क सूचना तथा डोमेन रजिस्ट्री की आवश्यकताओं के अनुरूप एक डोमेन नाम का उपयोग करने के दायित्वों से सम्बंधित. इसके अलावा प्रशासनिक संपर्क तकनीकी और बिलिंग कार्यों के लिए अतिरिक्त संपर्क जानकारी स्थापित करता है।
- *तकनीकी संपर्क.* तकनीकी संपर्क डोमेन नाम के नेम सर्वर का प्रबंधन संभालता है। एक तकनीकी सम्पर्क का कार्य डोमेन नाम का डोमेन रजिस्ट्री की आवश्यकताओं के साथ सेटअप, डोमेन ज़ोन रिकॉर्ड को बनाए रखना और नाम सर्वर को निरंतर कार्यशीलता प्रदान करना है *(*जिसके लिए डोमेन नाम तक पहुंच आवश्यक है*).*
- *बिलिंग/भुगतान सम्बंधित संपर्क.* डोमेन नाम रजिस्ट्रार से बिल प्राप्त करने और लागू फीस का भुगतान करने के लिए जिम्मेदार पार्टी यानि उपयोगकर्ता जो "पंजीकृत" या "डोमेन धारक" है।
- *नाम सर्वर.* ज्यादातर रजिस्ट्रार पंजीकरण सेवा के भाग के रूप में दो या अधिक सर्वर नाम प्रदान करते हैं। हालांकि, एक रजिस्ट्रेंट अपने आधिकारिक नाम सर्वर को एक डोमेन रिसोर्स रिकॉर्ड होस्ट करने के लिए निर्दिष्ट कर सकता है। रजिस्ट्रार की नीतियां सर्वरों की संख्या और सर्वर की आवश्यक सूचना के प्रकार पर नज़र रखती हैं। कुछ प्रदाताओं को एक होस्ट नाम और इसी से मिलते जुलते *IP* पते या केवल होस्टनाम की आवश्यकता होती है जो नए डोमेन में या तो ढूढने लायक होनी चाहिए अथवा उसका अस्तित्व कहीं और होना चाहिए। पारंपरिक आवश्यकताओं के आधार पर *(RFC 1034),* आम तौर पर दो सर्वर की एक न्यूनतम आवश्यकता पड़ती है।

.

दुरुपयोग और विनियमन

आलोचक अक्सर डोमेन नाम पर प्रशासनिक सत्ता के दुरुपयोग का दावा करते हैं। विशेष रूप से उल्लेखनीय *VeriSign* साइट खोजक प्रणाली है जो सभी अपंजीकृत *.Com* और *.NET* डोमेन को *VeriSign* वेबपेज पर भेज देती थी। उदाहरण के लिए, *VeriSign* के साथ एक सार्वजनिक सभा में *SiteFinder*[14] के बारे में तकनीकी चिंताओं पर बहस करते हुए, *IETF* और अन्य तकनीकी संस्थाओं में सक्रिय कई लोगों ने समझाया कि वे कैसे *VeriSign* द्वारा, इंटरनेट की बुनियादी सुविधाओं के एक प्रमुख घटक के बदलने से चकित थे, जिसने इसके लिए आम सहमति प्राप्त नहीं की थी। पहले पहल *SiteFinder* को, वेबसाइट से हर इंटरनेट पूछताछ के लिए प्रयुक्त किया गया और इसने धन कमाने के लिए प्रश्नों के लिए गलत डोमेन नाम का प्रयोग उपयोगकर्ता को *VeriSign* खोज साइट पर ले जा कर किया। दुर्भाग्य से, अन्य एप्लीकेशन जैसे कि ईमेल के कई प्रयोगों, प्रश्न के उत्तर की कमी के कारणों, ने एक संकेत दिया कि डोमेन मौजूद नहीं है और यह संदेश न भेजे सकने योग्य सन्देश के रूप में समझा जा सकता है। मूल *VeriSign* कार्यान्वयन ने मेल के लिए इस धारणा को तोड़ दिया, क्योंकि वह हमेशा गलत डोमेन नाम को *SiteFinder* पर भेज देते थे। जबकि *VeriSign* ने बाद में ईमेल के साथ *SiteFinder* का व्यवहार बदल दिया, फिर भी *VeriSign* द्वारा किये गये इस कृत्य के बारे में घोर विरोध प्रदर्शन किया गया कि कैसे *VeriSign* ने इंटरनेट संरचना घटक, जिसकी सुरक्षा का जिम्मेवार *VeriSign* था, के हित के बजाए अपने वित्तीय हितों की तरफ अधिक ध्यान दिया।

व्यापक आलोचना के बावजूद, *VeriSign* ने अनिच्छा से इसे हटाया जब नाम और नंबर प्रदान करने के लिए इंटरनेट कॉरपोरेशन *(ICANN)* ने रूट नाम सर्वर के प्रबंधन के अनुबंध को रद्द करने की धमकी दी। *ICANN* ने व्यापक संख्या में विचार विमर्श पत्र, समिति की रिपोर्ट और *ICANN* के निर्णय सार्वजनिक किए। [15]

ICANN पर अमेरिका के राजनीतिक प्रभाव के बारे में बेचैनी भी महत्वपूर्ण है। यह एक *.XXX* -शीर्ष स्तर डोमेन को बनाने के प्रयास के दौरान एक महत्वपूर्ण मुद्दा था और इसने वैकल्पिक *DNS* रूट की ओर ध्यान खींचा जो किसी एक देश के नियंत्रण से बाहर होगा। [16]

इसके अतिरिक्त, डोमेन नेम "फ्रंट रनिंग" सम्बंधित कई आरोप हैं, जिसके तहत रजिस्ट्रार को जब *whois* प्रश्न दिए जाते हैं, स्वचालित रूप से अपने लिए डोमेन नाम रजिस्टर कर लेते हैं। हाल ही में, नेटवर्क सोल्यूशन पर इस का आरोप लगाया गया है।[17]

डोमेन नाम अधिनियम का सत्य

संयुक्त राज्य अमेरिका में, डोमेन नाम में सत्य के अधिनियम *2003* व *2003* के रक्षा अधिनियम का संयोजन, भ्रामक डोमेन नामों का उपयोग बैन करता है जो लोगों को इंटरनेट पर अश्लील साहित्य युक्त साइटों पर जाने के लिए आकर्षित करते हैं।

इंटरनेट मानक

डोमेन नाम प्रणाली टिप्पणियों के लिए अनुरोध *(RFC)* द्वारा परिभाषित, इंटरनेट इंजीनियरिंग टास्क फोर्स द्वारा प्रकाशित *(*इंटरनेट मानक*)* दस्तावेज़ है। *RFCs* की निम्नलिखित सूची *DNS* प्रोटोकॉल को परिभाषित करती है।

- *RFC 920* , *डोमेन आवश्यकताएँ*- मूल शीर्ष स्तर के निर्दिष्ट डोमेन
- *RFC 1032* , *डोमेन एड्मिनिसट्रेटर गाइड*
- *RFC 1033* , *डोमेन एड्मिनिसट्रेटर ऑपरेशन गाइड*
- *RFC 1034* , *डोमेन नाम - विचार और सुविधाएं*
- *RFC 1035* , *डोमेन नाम - क्रियान्वन और विशिष्टताएं*
- *RFC 1101* , *नेटवर्क नामों और अन्य प्रकारों की डीएनएस एन्कोडिंग*
- *RFC 1123* , *इंटरनेट होस्ट के लिए आवश्यकताएँ-एप्लीकेशन और सहायता*
- *RFC 1178* , *अपने कंप्यूटर के लिए नाम का चयन करना (FYI 5)*
- *RFC 1183* , *नई DNS RR परिभाषाएं*
- *RFC 1591* , *डोमेन नाम प्रणाली संरचना और प्रतिनिधिमंडल (*जानकारी के लिए*)*
- *RFC 1912* , *सामान्य DNS परिचालनात्मक और सेटअप सम्बंधित त्रुटियाँ*

- *RFC 1995* , डीएनएस में इंक्रीमेंटल जोन का स्थानांतरण
- *RFC 1996* , जोन परिवर्तन की तत्काल सूचना के लिए एक तंत्र (*DNS NOTIFY*)
- *RFC 2100* , होस्ट का नामकरण (जानकारी)
- *RFC 2136* , डोमेन नाम प्रणाली में डाईनेमिक अपडेट (*DNS* अपडेट)
- *RFC 2181* , डीएनएस विशेषताओं का स्पष्टीकरण
- *RFC 2182* , माध्यमिक *DNS* सर्वर का चयन और प्रक्रिया
- *RFC 2308* , डीएनएस सम्बंधित प्रश्नों की नकारात्मक कैशिंग (*DNS NCACHE*)
- *RFC 2317* , वर्गहीन *IN-ADDR.ARPA* प्रतिनिधिमंडल (*BCP 20*)
- *RFC 2671* , *DNS* के लिए एक्सटेंशन प्रणाली (*EDNS0*)
- *RFC 2672* , नॉन-टर्मिनल डीएनएस नाम का पुनर्निर्धारण
- *RFC 3225* , *DNSSEC* की रिसोल्वर सहायता का संकेत देना
- *RFC 3226* , *DNSSEC* और *IPv6 A6* जानने वाले सर्वर/रिसोल्वर के संदेश के आकार की आवश्यकताएं
- *RFC 3597* , अज्ञात *DNS* संसाधन रिकॉर्ड (आर आर) प्रकारों की हैंडलिंग
- *RFC 3696* , नामों के जाँच और परिवर्तन के लिते एप्लीकेशन तकनीकें
- *RFC 4343* , डोमेन नाम प्रणाली (*DNS*) प्रकरण संवेदनहीनता के बारे में स्पष्टीकरण
- *RFC 4592* , डोमेन नाम प्रणाली में वाइल्डकार्ड की भूमिका
- *RFC 4892* , नाम सर्वर उदाहरण की पहचान के लिए एक तंत्र की आवश्यकताएँ (जानकारी)
- *RFC 5001* , *DNS* नाम सर्वर पहचानने के विकल्प (*NSID*)
- *RFC 5395* , डोमेन नाम प्रणाली (*DNS*) *IANA* सुझाव (*BCP 42*)

सुरक्षा

- *RFC 4033* , *DNS* सुरक्षा परिचय और आवश्यकताएँ
- *RFC 4034* , *DNS* सुरक्षा एक्सटेंशन्स के लिए संसाधन रिकार्ड
- *RFC 4035* , डीएनएस सुरक्षा एक्सटेंशन्स के लिए प्रोटोकॉल संशोधन

इन्हें भी देखें

- उच्चतम डोमेन
- डाईनेमिक *DNS*

- वैकल्पिक *DNS* रूट
- *DNS* सर्वर सॉफ्टवेयर की तुलना
- राउंड रोबिन *DNS*
- क्षितिज-विभाजित डीएनएस
- डीएनएस प्रबंधन सॉफ्टवेयर
- *DNS* कैशे विषाक्तता
- डीएनएस का अपहरण
- *DNS* रिकॉर्ड प्रकारों की सूची

सन्दर्भ

1. *Mockapetris, Paul (2004-01-02). "Letting DNS Loose" (https://web.archive.org/web/20100312062348/http://www.circleid.com/posts/letting_dns_loose) . CircleID. मूल (http://www.circleid.com/posts/letting_dns_loose/) से 12 मार्च 2010 को पुरालेखित. अभिगमन तिथि 24 फ़रवरी 2010.*
2. *RFC 3467 - डोमेन नाम प्रणाली (DNS) की भूमिका*
3. *"History of the DNS" (https://web.archive.org/web/20100527002935/http://www.lagunainternet.com/techsupport/history_of_dns.htm) . मूल (http://www.lagunainternet.com/techsupport/history_of_dns.htm) से 27 मई 2010 को पुरालेखित. अभिगमन तिथि 2008-04-29.*
4. *Cricket Liu, Paul Albitz. "DNS & BIND" (http://books.google.co.uk/books?id=zkZN52WhG8sC&pg=PA3&lpg=PA3&dq=sri+HOSTS.TXT&source=web&ots=wuZ79E-zJ2&sig=btF0Z2nclOnX_UgNj7a1f5S7Uqg&hl=en) . O'Reilly (shown via Google Books). मूल से 21 जुलाई 2011 को पुरालेखित (https://web.archive.org/web/20110721232444/http://books.google.co.uk/books?id=zkZN52WhG8sC&pg=PA3&lpg=PA3&dq=sri+HOSTS.TXT&source=web&ots=wuZ79E-zJ2&sig=btF0Z2nclOnX_UgNj7a1f5S7Uqg&hl=en) . अभिगमन तिथि 2008-04-29.*
5. *RFC 1034 , डोमेन नाम - विचार और सुविधाएं, पी. मोकापेट्रिस (नवंबर 1987)*
6. *RFC 1035 , डोमेन नाम - क्रियान्वन और विशेषताएं, पी. मोकापेट्रिस (नवम्बर 1987)*
7. *http://mydns.bboy.net/survey/ Archived (https://web.archive.org/web/20100407063925/http://mydns.bboy.net/survey/) 2010-04-07 at the Wayback Machine DNS Server Survey*
8. *एक डोमेन नाम की अधिकतम लंबाई क्या है? (http://www.ops.ietf.org/lists/namedroppers/namedroppers.2003/msg00964.html) Archived (https://web.archive.org/web/20100221225518/http://www.ops.ietf.org/lists/namedroppers/namedroppers.2003/msg00964.html) 2010-02-21 at*

the Wayback Machine IETF DNSOP कार्य समूह मेलिंग सूची पर. तार पर, DNS बाईनरी प्रारूप में, यह RFC के 1034 सेक्शन 3.1 के अनुसार यह अधिकतम 255 ओक्टेट्स हो सकता है। एक all-ASCII होस्टनाम के लिए, इसे पारंपरिक डॉट नोटेशन रूप में 253 अक्षर के रूप प्रदर्शित किया जा सकता है।

9. "Providers ignoring DNS TTL ?" (https://web.archive.org/web/20090519090354/http://ask.slashdot.org/article.pl?sid=05%2F04%2F18%2F198259) . Slashdot. 2005. मूल (http://ask.slashdot.org/article.pl?sid=05/04/18/198259) से 19 मई 2009 को पुरालेखित. अभिगमन तिथि 2009-01-03.
10. "How Internet Explorer uses the cache for DNS host entries" (http://support.microsoft.com/default.aspx?scid=KB;en-us;263558) . Microsoft. 2004. 263558. मूल से 29 जून 2011 को पुरालेखित (https://web.archive.org/web/20110629134145/http://support.microsoft.com/default.aspx?scid=KB;en-us;263558) . अभिगमन तिथि 2006-03-07.
11. Mockapetris, P (November 1987). "RFC 1035: Domain Names - Implementation and Specification" (http://www.ietf.org/rfc/rfc1035.txt) . मूल से 7 अप्रैल 2007 को पुरालेखित (https://web.archive.org/web/20070407140546/http://www.ietf.org/rfc/rfc1035.txt) . अभिगमन तिथि 24 फ़रवरी 2010.
12. RFC 4592 डोमेन नाम प्रणाली में वाइल्डकार्ड की भूमिका ई. लुईस (जुलाई 2006)
13. शब्द होस्ट नाम का प्रयोग यहाँ FQDN, के लिए एक होस्ट की तरह किया जा रहा है, जैसे उदाहरण के लिए En.wikipedia.org., और न कि केवल (इसी उदाहरण में) en के लिए.
यद्यपि अधिकतर डोमेन नाम वास्तव में होस्ट को निर्दिष्ट करते हैं, कुछ डोमेन नाम DNS प्रविष्टियों ऐसा नहीं भी कर सकती हैं। इस अर्थ में, एक (FQDN) होस्टनाम डोमेन नाम का एक प्रकार है, लेकिन सभी डोमेन नाम वास्तविक होस्ट नाम नहीं हैं। Cf. यह होस्ट नाम बनाम (http://www.ops.ietf.org/lists/namedroppers/namedroppers.2005/msg00889.html) Archived (https://web.archive.org/web/20070703195516/http://www.ops.ietf.org/lists/namedroppers/namedroppers.2005/msg00889.html) 2007-07-03 at the Wayback Machine डोमेन नाम स्पष्टीकरण (http://www.ops.ietf.org/lists/namedroppers/namedroppers.2005/msg00889.html) Archived (https://web.archive.org/web/20070703195516/http://www.ops.ietf.org/lists/namedroppers/namedroppers.2005/msg00889.html) 2007-07-03 at the Wayback Machine DNS OP IETF कार्य समूह से है।
14. McCullagh, Declan (2003-10-03). "VeriSign fends off critics at ICANN confab" (http://www.news.com/2100-1038-5088128.html) . CNET News.com. अभिगमन तिथि 2007-09-22.
15. Internet Corporation for Assigned Names and Numbers (ICANN). "Verisign's Wildcard Service Deployment" (https://www.icann.org/topics/wildcard-history.html) . मूल से 2 दिसंबर 2008 को पुरालेखित (https://web.archive.org/web/20081202125045/http://www.icann.org./topics/wildcard-history.html) . अभिगमन तिथि 2007-09-22.
16. Mueller, M (March 2004). Ruling the Root. MIT Press. आई॰ऍस॰बी॰ऍन॰ 0262632985.

17. *"स्लैशडॉट | NSI रजिस्टर प्रत्येक डोमेन की जाँच की गयी" (https://web.archive.org/web/20100217102228/http://slashdot.org/article.pl?sid=08%2F01%2F08%2F1920215)* . *मूल (http://slashdot.org/article.pl?sid=08%2F01%2F08%2F1920215)* *से 17 फ़रवरी 2010 को पुरालेखित. अभिगमन तिथि 14 जून 2020.*

बाहरी कड़ियाँ

- *domain name system computer networks (* डोमेन नेम सिस्टम*) (https://onlineaducation.com/domain-name-system-computer-networks/)*
- डीएनएस जटिलता, *(http://www.acmqueue.com/modules.php?name=Content&pa=showpage&pid=481)* पॉल विक्सी, *ACM* क्यू
- मुक्त स्रोत गाइड - रॉकेट वैज्ञानिकों के लिए डीएनएस, *(https://web.archive.org/web/20100217182048/http://www.zytrax.com/books/dns/)* आगे पढ़ने के लिए एक ऑन-लाइन तकनीकी पुस्तक
- कुछ *DNS* उपकरणों की एक सूची *(https://web.archive.org/web/20060902123229/http://www.bind9.net/)*
- सर्कल *ID* - डीएनएस से संबंधित सभी विषयों के लिए मुक्त समाचार और राय हब *(https://web.archive.org/web/20100224090452/http://www.circleid.com/topics/dns)*

Last edited 5 months ago by InternetArchiveBot

विकिपीडिया