Engineer

DNS

registret der gør et navn til en adresse

Hvad det er

DNS, Domain Name System, er det distribuerede register, internettet slår op i, før stort set alt andet sker. Et program med et navn som `example.com` spørger det om den adresse, der skal forbindes til, og det samme system svarer på en række andre spørgsmål om navnet: hvilke servere der tager imod dets mail, hvilken tekst ejeren har offentliggjort, og hvilke servere der har ansvaret for det.

Registret er et træ, der læses fra højre mod venstre. Øverst står roden, under den topdomæner som `.com` og `.dk`, under dem de domæner, folk registrerer, og under dem de underdomæner, ejerne selv opretter. Ingen enkelt server rummer hele træet. Hvert niveau overlader ansvaret for det næste til navngivne servere, og den del af træet, ét sæt servere svarer for, kaldes en zone. Den, der kontrollerer zonen, bestemmer, hvad dens navne peger på.

To slags servere deler arbejdet. Autoritative servere rummer en zone og giver det endelige svar for den. Rekursive resolvere, drevet af internetudbydere, virksomheder eller offentlige tjenester, går turen for klienten: de spørger en rodserver, hvor `.com` bor, spørger `.com`, hvor `example.com` bor, spørger domænets egne servere om posten og leverer svaret tilbage. Styresystemet på din maskine har kun en lille stub-resolver, der sender spørgsmålet videre til en rekursiv resolver og venter.

Svarene er typede poster, og en zonefil er den klassiske måde at skrive dem ned på:

```
$ORIGIN example.com.
$TTL 3600
@     IN  SOA    ns1.example.net. hostmaster.example.com. (
                 2026101701 7200 900 1209600 300 )
```

```
@     IN  NS     ns1.example.net.
@     IN  A      192.0.2.10
@     IN  AAAA   2001:db8::10
www   IN  CNAME  example.com.
@     IN  MX     10 mail.example.net.
@     IN  TXT    "v=spf1 include:_spf.example.net -all"
```

`A` og `AAAA` giver IPv4- og IPv6-adresser. `CNAME` gør ét navn til alias for et andet. `MX` angiver den server, der modtager mail, og `TXT` bærer fri tekst, hvilket i praksis vil sige politikker for mailgodkendelse og beviser på, at nogen kontrollerer domænet. `NS` og `SOA` beskriver selve zonen.

Hver post har en levetid, en TTL, her 3.600 sekunder, og det tal er nøglen til, hvordan DNS opfører sig. En resolver, der har hentet et svar, må genbruge det, indtil TTL'en udløber, uden at spørge igen. Resultatet er et system, der svarer på de fleste opslag fra en cache tæt på klienten og bliver ved med at svare, når en autoritativ server kortvarigt er væk, til gengæld for at ingen kan tvinge verden til at glemme et gammelt svar før tid.

Brug begrebet i en beslutning

Brug »Hvad er DNS?« i en konkret beslutning. Vi kan knytte begrebet til det system, I bygger eller køber. Wi-Fi, firewalls og fjernadgang sat op, så medarbejdere kan arbejde hvor som helst, og kun de rette kommer ind.

Se vores ydelser Kontakt os →

Undersøg antagelserne bag ordet

Brug forklaringen og det tilknyttede arbejde til at formulere spørgsmål til jeres projekt. Vi kan undersøge antagelserne sammen, før et værktøj vælges. Har administreret 30 Ubuntu Linux VPS-instanser — 30 Linux-servere driftet, med en plan for gendannelse

Har administreret 30 Ubuntu Linux VPS-instanser Kontakt os →

Hvorfor det betyder noget

Enhver forbindelse til en navngiven tjeneste begynder med et opslag. Er DNS forkert, kan serverne være helt sunde, og alligevel ser sitet, API'et og mailen ud til at være nede, fordi intet når frem til dem. Er DNS rigtigt, lægger ingen mærke til det, og derfor er det ofte den dårligst dokumenterede del af et system.

DNS er også stedet, hvor kontrollen med et domæne udøves og bevises. Mailudbydere læser SPF-, DKIM- og DMARC-politikker fra `TXT` -poster for at afgøre, om en besked, der påstår at komme fra dit domæne, er ægte, så en forkert eller manglende post sender rigtig mail i spamfilteret. Certifikatudstedere og hostede tjenester beder om en post med en bestemt

værdi som bevis på, at du ejer navnet. Den, der kan redigere zonen, kan omdirigere sitet, opsnappe mailen og få udstedt certifikater til domænet, og derfor er adgangen til DNS-kontoen lige så følsom som adgangen til serverne.

Caching præger hver ændring. Det, folk kalder udbredelse, er i virkeligheden tusindvis af resolvere, der hver venter på, at deres egen kopi udløber. Vil man flytte et site med mindst mulig forstyrrelse, sænker man TTL'en et døgn eller så i forvejen, venter, til den gamle TTL er gået, laver ændringen og hæver TTL'en igen, når alt er faldet på plads.

Klassisk DNS sendes som klartekst, og hverken svarene eller spørgsmålene er private. DNSSEC lader en resolver kontrollere, at et svar virkelig kommer fra zonens ejer og ikke er ændret undervejs, og ifølge sin egen specifikation giver det ingen fortrolighed. Krypterede transporter som DNS over HTTPS skjuler spørgsmålene for nettet imellem, mens resolveren i den anden ende stadig ser hvert eneste navn, der bliver slået op. At vælge resolver er at vælge, hvem der får den liste.

Sådan ser det ud i praksis

Virksomhedens egen zone viser det mest almindelige udgangspunkt og en vej ud af det. Den rummede omkring tyve poster, der lå i en udbyders webkonsol, og de dækkede webtilstedeværelsen, et git-underdomæne, to personlige omdirigeringer, mail sendt gennem en hostet postkasseudbyder, et domæne til transaktionsmails og godkendelsesposterne for begge. I en konsol er den aktuelle tilstand det, den sidste person, der klikkede, efterlod. Nu er zonen data i et repository: aktive poster, applikationsposter, poster til transportpolitik og en særskilt liste over poster, der står for at blive fjernet. Ét play afstemmer erklæringen mod udbyderens API. Bag eksplicitte tilvalgsmærker ligger yderligere to indgange: en audit, der viser forskellen uden at røre noget, og en eksport, der skriver zonen ud i det standardiserede zonefilformat, så den kan læses af andet end repositoryet. Auditten er den del, der bruges mest, for oftere vil man kende forskellen end lukke den.

Samme udbyder har en bevidst smal opgave. Sitets kolofon forklarer, at Cloudflare besvarer DNS for sitet og intet andet, uden en proxy foran, og den skriver serverens adresse lige ud, fordi DNS alligevel udleverer den adresse til enhver, der spørger.

At læse DNS hører også med til den faste beskrivelse af driftsmiljøet. Blandt syv rapportroller, der kun læser og gengiver den kørende vært som Markdown, findes en udbyderoversigt, der læser DNS side om side med to cloud-API'er og et API til dedikerede servere, under reglen om, at ingen rapport skriver et tal, som kørslen ikke selv har målt.

Hvor det går galt

DNS-fejl er ofte forsinkede, cachede eller delte, så det samme navn virker for den ene og svigter for den anden. Disse kommer igen og igen.

En ændring vil ikke slå igennem. Nogen retter en post med en TTL på et døgn og forventer, at den nye adresse gælder med det samme, men resolvere, der har det gamle svar i cachen, bliver ved med at levere det, til deres kopi udløber. En stillere udgave rammer nye navne: blev navnet slået op, før posten fandtes, caches det negative svar også, så længe zonens `SOA` tillader det (RFC 2308). Sænk TTL'en før planlagte ændringer, og opret poster, før noget spørger efter dem.

Et alias står et sted, hvor det ikke må. RFC 1034 siger, at et navn med en `CNAME` ikke må have andre data. Det nøgne domæne, selve `example.com`, har altid `SOA`- og `NS`-poster, så en zone, der følger standarden, kan ikke gøre roden til et alias, og en `CNAME` på et underdomæne, der også skal have `MX`- eller `TXT`-poster, bryder samme regel. Udbydere tilbyder fladtrykte poster eller `ALIAS` som deres egen løsning; de opfører sig hver lidt forskelligt, så undersøg, hvad din udbyder faktisk leverer.

Mailgodkendelsen vælter. Hvert `include` i en SPF-post udløser DNS-opslag, og RFC 7208 sætter loftet for det samlede antal ved ti. Én ekstra nyhedsbrevs- eller supporttjeneste kan skubbe posten over grænsen, og så behandler modtagerne politikken som en permanent fejl, og den mail, den skulle godkende, bliver vurderet uden den. Tæl opslagene, hver gang en afsender kommer til.

En post overlever sit mål. Et underdomæne peger med `CNAME` på en lagerbeholder i skyen eller en hostet app, der siden er slettet. Lader udbyderen hvem som helst gøre krav på ressourcenavnet, kan en fremmed overtage det og servere sit eget indhold under dit underdomæne, med dit domænes omdømme hæftet på. Fjern posten i den samme ændring, som fjerner det, den peger på.

Zonen findes kun i en konsol. Der er ingen historik, ingen gennemgang og ingen diff, så en fejlagtigt slettet `MX` -post opdages, når mailen holder op med at komme, og den tidligere værdi må stykkes sammen efter hukommelsen. En jævnlig eksport af zonen, eller bedre en erklæring som kode, giver hver ændring et spor og en vej tilbage.

Læs mere

- RFC 1035: Domain Names, Implementation and Specification — postformaterne, zonefilens syntaks og protokollen på ledningen.
- RFC 1034: Domain Names, Concepts and Facilities — træet, zoner, delegering og hvordan resolvere bevæger sig gennem dem.
- RFC 9499: DNS Terminology — omhyggelige definitioner af de ord, denne side bruger.
- RFC 2308: Negative Caching of DNS Queries — hvor længe en resolver må huske, at et navn ikke findes.
- RFC 4033: DNS Security Introduction and Requirements — hvad DNSSEC beskytter, og hvad det lader være.
- RFC 8484: DNS Queries over HTTPS — opslag pakket ind i almindelig krypteret webtrafik.

Ordliste

Vælg ud fra jeres faktiske vilkår

Medbring udrulningsforløbet, driftsrisiciene og dem, der har vagten. Vi kan afgrænse en ændring med kontroller, en vej tilbage og tydeligt ansvar.

Se, hvad det koster Kontakt os →

Gør begreberne praktiske

Spørg, hvordan »Hvad er DNS?« gælder i jeres organisation. Et konkret eksempel fra jeres nuværende system er nok til at begynde samtalen.

Se vores ydelser

welcome@engineer.company +45 50 10 61 84 København, Danmark

Tag denne side med dig

Visitkort

Lavet med kærlighed af D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – nu · Engineer ApS · Alle rettigheder forbeholdes.

Udskrevet fra https://engineer.company/da/glossary/dns/