Engineer

E-post

post sendt mellom postkasser over internett

Hva det er

Bak hver adresse på formen `navn@domene` står e-post, systemet som frakter en skrevet melding fram til adressen og legger den i en postkasse til eieren leser den. På engelsk skrives det email eller e-mail, og standardene som definerer det, sier for det meste bare mail. Mottakeren trenger ikke være pålogget når meldingen sendes: servere tar imot meldingen, holder på den og sender den videre, og derfor beskrives e-post som lagre-og-videresend (store-and-forward) og ikke som en samtale.

To dokumenter gjør det meste av definisjonsarbeidet. RFC 5322 sier hva en melding er: en blokk med headerfelt, én tom linje og deretter innholdet. RFC 5321 beskriver SMTP, protokollen en sendende server bruker for å gi meldingen videre til neste server på veien. For å finne den neste serveren slår avsenderen opp domenets MX-poster (mail exchanger) i DNS, og de navngir maskinene som er villige til å ta imot post for domenet.

En komplett melding kan være svært kort:

```
From: Ana Holm <ana@example.org>
To: orders@example.com
Date: Fri, 16 Oct 2026 09:14:00 +0200
Subject: Invoice 114 has the wrong VAT number
Message-ID: <20261016.0914.ana@example.org>

Hello, could you send a corrected copy?
```

Bare to av de headerfeltene er påkrevd: datoen og adressen til opphavspersonen. Alt annet er valgfritt etter standardens bokstav, selv om en melding uten emne eller meldings-ID ser slurvete ut for programmene som

sorterer post. En linje kan være på høyst 998 tegn og bør holde seg innenfor 78, fordi mange e-postprogrammer fortsatt kutter eller bryter alt som er lengre.

Under dette ligger et andre, skjult lag. SMTP flytter meldingen inne i en *konvolutt*: en returadresse, som rapporter om feil sendes til, og en liste med mottakere. Konvolutten og linjene `From:` og `To:` som leseren ser, er atskilte og er ofte forskjellige. En blindkopi virker fordi mottakeren står på konvolutten og mangler i headerne. En e-postliste virker fordi konvoluttens returadresse peker på listen, slik at en returmelding fra et abonnement som er dødt, når listeprogramvaren og ikke den som skrev innlegget.

Bruk begrepet i en beslutning

Bruk «Hva er e-post?» i en konkret beslutning. Vi kan knytte begrepet til systemet dere bygger eller kjøper. Et nettsted som er raskt, lett å finne og kan redigeres av dine egne ansatte, uten en regning hver gang en formulering endres.

Se tjenestene våre Ta kontakt →

Undersøk antakelsene bak ordet

Bruk forklaringen og det tilknyttede arbeidet til å formulere spørsmål for prosjektet. Vi kan undersøke antakelsene sammen før et verktøy velges. Utviklet 100 webapplikasjoner med HTML/HTML5 — 100 webapplikasjoner bygd

Utviklet 100 webapplikasjoner med HTML/HTML5 Ta kontakt →

Hvorfor det er viktig

En e-postadresse er identiteten det meste av weben faller tilbake på. Bekreftelser ved registrering, tilbakestilling av passord, kvitteringer, fakturaer og beskjeden om at en betaling er avvist, går alle med e-post. Et produkt der e-postene forsvinner, har brukere som er låst ute uten at noen merker det. Den som aldri fikk bekreftelseslenken sin, skriver sjelden for å si fra. Vedkommende går.

E-post har heller ingen eier. Hvert domene kan drive sine egne servere, og hver server kan snakke med hvilken som helst annen, og den åpenheten er grunnen til at e-post har overlevd nesten hver meldingstjeneste som er bygd siden. Prisen er tillit. Det opprinnelige designet lar en avsender skrive hva som helst på linjen `From:`, så mottakende servere avgjør nå selv om de skal tro på en melding. De avgjør det ut fra bevis som det sendende domenet publiserer i DNS: hvilke verter som får sende for det, hvilke nøkler som signerer posten, og hva som skal skje med en melding som utgir seg for å komme fra domenet og verken har det ene eller det andre.

For en virksomhet går det nyttige skillet mellom to slags trafikk. Transaksjonell e-post går til én person fordi den personen nettopp gjorde noe: en kode, en kvittering, en lenke for tilbakestilling. Masseutsendelser går til mange fordi avsenderen valgte å sende dem. De to bygger opp omdømme hver for seg, får klager i ulikt tempo og sendes ofte fra forskjellige domener, så blander du dem, kan et nyhetsbrev som irriterer folk, forsinke tilbakestillingen av et passord.

Til slutt er det spørsmålet om hvor posten bor. Å drive en e-postserver betyr datalagring, filtrering av søppelpost, oppdateringer og et avsenderomdømme som må opparbeides og holdes ved like. En hostet postkasse til å lese i og en leverandør for automatisk utsending er det vanlige svaret, og det er et valg som er verdt å ta med vilje i stedet for å la det skje av seg selv.

Slik ser det ut i praksis

På én plattform bar e-post verifisering, varslinger, sammendrag og kampanjer, og med én enkelt leverandør ble hvert driftsavbrudd totalt. Svaret var å sette tre leverandører bak ett grensesnitt med automatisk failover: SendGrid som primær, deretter SMTP2GO og Azure Communication Services. Innkommende webhooks rapporterer hva hver leverandør gjorde med en melding, og begge sider havner i en sendelogg og en tabell for leveringshendelser, slik at leveringsstatusen leverandøren rapporterer, kan hentes med en spørring i stedet for å gjettes. Kampanjer til brukersegmenter går gjennom et eget skjema for masseutsendelser med 5 tabeller og 24 funksjoner, bygd atskilt fra transaksjonell e-post fordi det er et annet problem. En oppstartskontroll sender en ekte melding gjennom hele stakken, styrt av et flagg. I utvikling logger en feil en advarsel, og i staging og produksjon stopper den prosessen. Siden nevner også prisen: tre integrasjoner å holde i gang, og leveringslogger som vokser og må ryddes.

Selskapets egen e-post begynner i DNS. Sonen har omtrent tjue poster, blant dem e-postruting gjennom en leverandør av hostede postbokser, et domene for transaksjonell utsending og autentiseringspostene for begge, og hele sonen holdes som data i repoet og avstemmes mot leverandørens API. Et eget inngangspunkt for revisjon rapporterer forskjellen mellom deklarasjonen og det som faktisk serveres, uten å endre noe, og det er den delen som brukes mest.

Hvor selskapets egen e-post skal bo, ble avgjort med regnestykker. De offentlige tjenestene kjører på én vert med 512 MB minne, og e-post holdes med vilje borte fra den maskinen: en e-postserver trenger margin som verten allerede har brukt

opp, så den er skrevet som et installasjonsstillas som venter på en egen vert. I mellomtiden leses brev til selskapet i selskapets egen postkasse, hostet hos Apple iCloud, og det står ingen e-postliste bak den.

Hvor det går galt

Posten havner i søppelposten den uka leverandøren byttes. Applikasjonen er uendret, men mottakerne blir likevel mistenksomme. Domenets publiserte poster nevner fortsatt den gamle avsenderen, så post fra den nye kommer fram usignert eller fra en vert domenet aldri har godkjent. Legg den nye avsenderen inn i postene før byttet, behold den gamle til den siste meldingen derfra er sendt, og les de samlede rapportene mottakerne sender tilbake, hvis domenet ber om dem.

Applikasjonen starter, og ingen får noe. En utløpt nøkkel eller en suspendert senderkonto ser nøyaktig ut som en konfigurasjon som virker, helt til et ekte menneske venter på en kode. Send en ekte melding ved oppstart der stillhet er farlig, og registrer hva leverandøren sier skjedde med hver melding, slik at en savnet e-post kan spores i stedet for å diskuteres.

Returmeldinger hoper seg opp på en adresse som ingen leser. Konvoluttens returadresse peker på en `no-reply` -boks, så permanente feil kommer inn og blir liggende uåpnet. Avsenderen fortsetter å skrive til postkasser som ble lagt ned for et år siden, og mottakerne leser det som vanen til en spammer. La returadressen peke på noe som behandler rapportene, og slutt å sende til en adresse etter en permanent feil.

Et midlertidig avslag blir behandlet som endelig. SMTP-svar som begynner med 4, betyr prøv igjen senere, og svar som begynner med 5, betyr stopp. En avsender som gir opp ved første 4xx, mister post som en travel eller forsiktig server ville ha tatt imot en halvtime senere. RFC 5321 ber om nye forsøk med minst 30 minutters mellomrom og om at avsenderen tidligst gir opp etter fire til fem dager. Transaksjonell e-post vil som regel ha raskere nye forsøk med backoff, og en tydelig registrering når avsenderen til slutt slutter å prøve.

Les mer

- RFC 5322: Internet Message Format — headerfeltene, den tomme linjen, grensene for linjelengde og de to feltene hver melding må ha.

- RFC 5321: Simple Mail Transfer Protocol — konvolutten, svarkodene, oppslaget av MX-poster og rådene om nye forsøk.
- RFC 7208: Sender Policy Framework (SPF) — hvordan et domene lister opp vertene som får sende e-post i dets navn.
- RFC 6376: DomainKeys Identified Mail (DKIM) Signatures — signering av headere og innhold med en nøkkel mottakeren henter fra DNS.
- RFC 7489: DMARC — knytter begge sjekkene til domenet i den synlige linjen `From:` , med en policy og rapporter; et informativt dokument og ikke en standard.

Ordliste

Velg ut fra deres faktiske rammer

Beskriv arbeidsflyten, eksisterende grensesnitt og brukerne som trenger endringen. Vi kan definere en testbar leveranse og en overlevering som gjør teamet i stand til å vedlikeholde løsningen.

Se hva det koster Ta kontakt →

Gjør begrepene praktiske

Spør hvordan «Hva er e-post?» gjelder i organisasjonen deres. Et konkret eksempel fra dagens system er nok til å starte samtalen.

Se tjenestene våre

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

Ta med deg denne siden

Visittkort

Laget med kjærlighet av D.

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

Opphavsrett © 2025 – nå · Engineer ApS · Alle rettigheter forbeholdt.

Skrevet ut fra https://engineer.company/nb/glossary/email/