Engineer

Onion-tjänst

en adress som bara fungerar via Tor

Vad det är

En onion-tjänst är en server som läsare når inne i Tor-nätverket via ett namn som bygger på serverns egen publika nyckel, i stället för ett DNS-namn som pekar på en IP-adress. I vardagligt tal kortas det ofta till en onion, en och samma tor-daemon kan köra flera onion-tjänster sida vid sida, och specifikationen använder fortfarande det äldre namnet hidden service.

Namnet består av 56 tecken base32 följda av `.onion`. Avkodat blir det 35 byte: tjänstens 32 byte långa identitetsnyckel av typen ed25519, en kontrollsumma på två byte tagen ur en SHA3-256-hash av nyckeln och en versionsbyte satt till 3. Eftersom den byten kommer sist slutar varje aktuell adress på bokstaven d. Etiketten väljs alltså inte av någon; den är nyckeln utskriven. Namnen i version 2 som kom före var 16 tecken hämtade ur en SHA-1-hash av en RSA-1024-nyckel, och tor tog bort dem i version 0.4.6.1-alpha, 2021.

Att nå tjänsten kräver två mötesplatser. Tjänsten väljer några reläer som introduktionspunkter, håller kretsar öppna till dem och laddar upp en signerad deskriptor som listar dem till katalogreläer. Deskriptorn arkiveras under en förblindad nyckel som byts varje period, och innehållet är krypterat, så ett relä som lagrar den utan att känna till adressen kan inte läsa listan. En klient som har adressen hämtar deskriptorn, kontrollerar signaturen mot nyckeln i namnet, väljer ett relä som mötespunkt och ber tjänsten via en introduktionspunkt att mötas där. Tjänsten bygger en egen krets till det reläet och slutför handskakningen, och reläet kopplar ihop de två kretsarna. Vägen passerar sex reläer, tre valda av vardera sidan, och ingen av ändarna får veta den andras IP-adress. Ett valfritt läge, som numera heter restricted discovery och tidigare hette client authorization, krypterar dessutom deskriptorn för varje behörig klients nyckel, så att bara de klienterna får veta var introduktionspunkterna finns.

Använd begreppet i ett beslut

Använd ”Vad är en onion-tjänst?” i ett konkret beslut. Vi kan knyta begreppet till systemet ni bygger eller köper. Wi-Fi, brandväggar och fjärråtkomst uppsatta så att personalen kan arbeta varifrån som helst och bara rätt personer kommer in.

Se våra tjänster Hör av er →

Granska antagandena bakom ordet

Använd förklaringen och det länkade arbetet för att formulera frågor till projektet. Vi kan undersöka antagandena tillsammans innan ett verktyg väljs. Förvaltade 30 VPS-instanser med Ubuntu Linux — 30 förvaltade Linux-servrar, med en plan för återställning

Förvaltade 30 VPS-instanser med Ubuntu Linux Hör av er →

Varför det spelar roll

Namnet gör det jobb som DNS och en certifikatutfärdare delar på för en vanlig webbplats, och behöver ingen av dem. Att deskriptorn verifieras och handskakningen slutförs visar klienten att den talar med den som har den privata nyckeln bakom adressen. RFC 7686, som reserverar `.onion` som ett domännamn för särskilt bruk, beskriver sådana namn som en förening av plats och autentisering och drar slutsatsen: ett onion-namn kan inte registreras, tilldelas, överlåtas eller återkallas. Att äga ett innebär att kontrollera nyckelparet, ingenting mer.

Krypteringen kommer ur samma konstruktion. Trafik till en onion-tjänst lämnar aldrig Tor-nätverket, så det finns inget utgångsrelä i andra änden och ingen sista sträcka i klartext. Vanlig `http://` till en onion bär därför en sida som är krypterad hela vägen, till en server vars identitet klienten har kontrollerat mot det namn den skrev in.

Tjänsten ringer bara ut, till sina introduktionspunkter och till varje mötespunkt, så den behöver ingen öppen inkommande port och fungerar bakom NAT. Att lägga till en onion lämnar listan över öppna portar på maskinen precis som den var.

Läsarna måste ändå få veta adressen. En webbplats kan tala om den med ett `Onion-Location` -huvudfält i svaret, och Tor Browser visar då en lila ruta i adressfältet som leder till onion-kopian av sidan. Tor Project ställer tre villkor: värdet är en `http:` - eller `https:` -URL vars värdnamn är en onion-adress, sidan som skickar det kom över HTTPS, och den sidan ligger inte själv på en onion. En meta-tagg kan bära samma besked där huvudfält inte går att sätta, men utan sökvägen, så varje sida pekar på en och samma fasta URL.

Så ser det ut i praktiken

Hur företagets onion-spegel publicerades under en läsbar adress beskrivs steg för steg. Daemonen kommer från distributionens eget paket, med den utgående proxyporten avstängd. Den privata nyckeln bakom adressen genererades utanför servern och har aldrig nått repot eller en bärbar dator; rollen avstår från att skriva över en identitet som den inte själv har genererat, så en nyckel som placerats för hand används som den är. Adressen är ett deklarerat värde i inventory, och varje konvergenskörning kontrollerar att identitetsfilen på servern stämmer med det, så en utbytt nyckel med en inaktuell deklaration stoppar körningen. Båda spärrarna mot hemligheter fångar nyckelfilen, sedan en mätning hade visat att det självklara mönstret för nyckelfiler matchade en deploy-nyckel och missade just den här.

En läsbar början kan hittas men aldrig väljas. Varje extra tecken multiplicerar den förväntade sökningen med 32, så de åtta bokstäverna i `engineer` kräver i snitt omkring 1,1 biljoner nyckelpar. Den slumpmässiga adress som spegeln först levererades med pensionerades i en stegvis övergång, så den publicerade adressen fortsatte att svara hela tiden. Ett steg såg slutgiltigt ut men var det inte: nyckeln kom på plats, konvergenskörningen rapporterade ingen ändring, och den nya adressen förblev opublicerad eftersom ingenting namngav dess katalog. Rollen fallerar nu på varje identitet på disk som ingen tjänst namnger.

Kolofonen skriver ut hela adressen, `engineer2ezpeji52ipphljcflzyhetvy5r6tnqubtq45nzlxxv6rsqd.onion`, bredvid adresserna för Gemini och Gopher. Alla fyra ytterdörrar läser ur samma katalog, så en enda driftsättning uppdaterar dem alla, och den vanliga webbplatsen skickar `Onion-Location`, så att Tor Browser erbjuder onion-adressen utan att läsaren behöver be om den. Sidan redogör också för vad läsarens eget nätverk ser genom varje dörr: över Gopher vilken sida som läses, över Gemini vilken adress som efterfrågades, över onion ingetdera. Den kallar prefixet en bekvämlighet snarare än ett bevis och ber läsarna jämföra hela adressen.

Adressens struktur går att kontrollera offline. De här raderna avkodar den och räknar om kontrollsumman, och båda jämförelserna skriver ut `True`:

```
import base64
import hashlib

label = "engineer2ezpeji52ipphljcflzyhetvy5r6tnqubtq45nzlxxv6rsqd"
raw = base64.b32decode(label.upper())
```

```
key, checksum, version = raw[:32], raw[32:34], raw[34:]
digest = hashlib.sha3_256(b".onion checksum" + key + version).digest()
print(version == b"\x03", digest[:2] == checksum)
```

En giltig kontrollsumma visar att adressen är korrekt formad, inte vems den är. Den delen kommer från att läsa den på företagets egna HTTPS-sidor.

Var det går fel

Nyckeln är den enda lagfarten. Utan någon registrar bakom namnet betyder en förlorad privat nyckel en förlorad adress, och en kopierad nyckel låter någon annan svara på den. Installationsguiden från Tor Project säger att nyckeln ska hållas privat och säkerhetskopieras om tjänsten är tänkt att leva länge. Behandla den som en signeringsnyckel, inte som konfiguration.

Att bara läsa de första bokstäverna. RFC 7686 varnar för att en angripare kan hitta nycklar vars namn liknar det avsedda, och att risken är större eftersom namnen inte betyder något för en människa. Den räknar läsbara namn, köpta med betydande beräkningskraft, som ett botemedel, men ett prefix på åtta tecken lämnar de övriga 48 lika ogenomskinliga som förut. Jämför hela adressen, eller kom dit via ett bokmärke eller en länk på webbplatsens egna HTTPS-sidor.

Servern namnger sig själv. Tor döljer var tjänsten körs, men programvaran bakom den kan fortfarande avslöja det. RFC 7686 räknar en server som röjer sin identitet i ett meddelande på applikationsnivå till de sätt som skyddet kan fallera på, och en felsida som skriver ut maskinens verkliga värdnamn är ett sådant. Installationsguiden rekommenderar också att trafiken skickas vidare till webbservern över en Unix-socket i stället för TCP, så att tjänsten inte läcker ut på det lokala nätverket:

```
HiddenServiceDir /var/lib/tor/my-website/
HiddenServicePort 80 unix:/var/run/tor/my-website.sock
```

Det spelar störst roll där själva platsen är det hemliga.

Namnet går till DNS. Programvara som inte talar Tor frågar den vanliga resolvern om ett onion-namn, vilket berättar för det lokala nätverket och DNS-leverantören vilken onion som efterfrågades. RFC 7686 säger att sådan

programvara ska misslyckas utan att göra uppslaget, och att DNS-servrar ska svara att namnet inte finns. Lämna i stället namnet till Tor-klienten.

En `Onion-Location` -ruta som aldrig syns. Ett huvudfält som bara sätts på omdirigeringen över vanlig HTTP bryter mot villkoret om HTTPS, och ett med en felstavad adress skickar läsarna ingenstans. Kontrollera huvudfältet på det slutliga HTTPS-svaret.

Läs mer

- Tor Rendezvous Specification, version 3 — hela protokollet: identitetsnycklar och förblindade nycklar, deskriptorer, introduktion och rendezvous.
- Encoding onion addresses — formeln som gör om en publik nyckel till ett namn på 56 tecken, med exempeladresser.
- How do Onion Services work? — en genomgång av introduktionspunkter, deskriptorer och mötet, steg för steg.
- Set up Your Onion Service — raderna i torrc, det som tor skapar på disk och råd om nycklar och om läckor av plats.
- Onion-Location — inställningar med huvudfält och meta-tagg för Apache, Nginx och Caddy, och villkoren som huvudfältet måste uppfylla.
- RFC 7686 — varför `.onion` är reserverat och vad resolvrar, DNS-operatörer och registrarer måste göra med det.

Ordlista

Välj utifrån era faktiska förutsättningar

Ta med driftsättningsvägen, driftsriskerna och dem som har jour. Vi kan avgränsa en förändring med kontroller, en väg tillbaka och tydligt ansvar.

Se vad det kostar Hör av er →

Gör begreppen praktiska

Fråga hur ”Vad är en onion-tjänst?” gäller i er organisation. Ett konkret exempel från ert nuvarande system räcker för att börja samtalet.

Se våra tjänster

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

Ta med dig den här sidan

Visitkort

Omsorgsfullt skapad med kärlek av D.

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

Upphovsrätt © 2025 – nu · Engineer ApS · Alla rättigheter förbehållna.

Utskrivet från https://engineer.company/sv/glossary/onion/