Engineer

Firewall

regelsættet, der afgør, hvilken trafik slipper igennem

Hvad det er

En firewall er et filter i netværkstrafikkens vej, som sammenligner hver pakke eller forbindelse med en liste af regler og derefter lader den passere, kasserer den eller sender et afslag tilbage. Den kan være software på den maskine, den beskytter, en selvstændig boks mellem to netværk eller en tjeneste, som en cloududbyder kører foran en virtuel maskine. Linux-folk taler ofte om et pakkefilter, og hos cloududbyderne hedder firewallen nogle steder en security group. Firewalls kommer typisk i lag: én hos udbyderen og én på selve værten, og firewallene behøver ikke at være skrevet i samme værktøj.

NIST's vejledning, SP 800-41 Rev. 1 fra september 2009, inddeler dem efter, hvor meget af trafikken de læser. Et pakkefilter ser på adresser, protokol og portnumre, én pakke ad gangen. En stateful firewall husker desuden forbindelserne, så svaret på en anmodning, serveren selv har sendt, kommer ind uden en regel for sig selv. En application-proxy gateway afslutter forbindelsen og læser protokollen indeni. En værtsbaseret firewall kører på den server eller bærbare, den vogter, og ved, hvilket program der ejer en socket.

På en Linux-server foregår filtreringen i kernens netfilter-framework. nftables, som har ligget i kernen siden version 3.13, afløser de ældre iptables-værktøjer, og frontends som ufw skriver reglerne for dem, der hellere vil udtrykke en hensigt end bygge kæder. Uanset hvad der skriver dem, er modellen den samme: en ordnet liste, der læses oppefra, hvor den første regel, der matcher, bestemmer udfaldet.

Firewallen forveksles tit med sine naboer. En web application firewall læser HTTP-anmodninger og blokerer angrebsmønstre i dem, hvor en netværksfirewall holder sig til adresser og porte. Et værktøj, der spærrer adresser efter mistænkelig adfærd, læser logfiler og tilføjer midlertidige regler mod dem og er

derfor afhængigt af en firewall nedenunder. En VPN afgør, hvem der overhovedet er på netværket. De svarer på hver sit spørgsmål, og regellisten står tilbage som fundamentet under dem alle.

Brug begrebet i en beslutning

Brug »Hvad er en firewall?« i en konkret beslutning. Vi kan knytte begrebet til det system, I bygger eller køber. Konti, adgangskoder og rettigheder sat op, så folk har præcis den adgang, jobbet kræver, og fratrådte mister den samme dag.

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 skrevet 1.000 RBAC-regler til Linux-servere — 1,000 adgangsregler skrevet for at lukke et sikkerhedshul

Har skrevet 1.000 RBAC-regler til Linux-servere Kontakt os →

Hvorfor det betyder noget

Hver tjeneste, der lytter på en offentlig adresse, er en dør, og firewallen er fortegnelsen over, hvilke døre der findes. Beslutningen har tre dele: hvor firewallen bor, hvad den tillader som udgangspunkt, og hvem der må ændre den.

En firewall hos udbyderen stopper trafikken, før den når maskinen, virker stadig, hvis nogen får root på værten, og koster ikke serveren noget CPU. Dens regler ligger i en konsol eller et API uden for serverens egen konfiguration, så de glider fra virkeligheden, medmindre de skrives som kode. En firewall på værten følger maskinen, kan begrænse en enkelt port og kan testes fra maskinen selv, men en kompromitteret root-konto kan omskrive den, og det kan software, der styrer sine egne regler, også. At køre begge er almindeligt og fornuftigt, og prisen er to lister, der skal holdes i takt: hver port, der åbnes eller lukkes, er en ændring to steder.

Udgangspunktet er den anden del. NIST anbefaler at afvise al trafik, der ikke udtrykkeligt er tilladt, og for indgående trafik er stort set alle enige. Udgående filtrering er der, hvor kompromiset mærkes: den begrænser, hvad en kompromitteret tjeneste kan nå, og den stopper pakkeopdateringer, fornyelse af certifikater og overvågning, indtil hver destination har fået sin regel.

Den tredje del er ejerskab. En firewall med default deny er kun så korrekt som listen over de porte, nogen har erklæret, så det værdifulde er netop den erklærede liste, holdt i versionsstyring og kontrolleret mod de aktive regler. Softwaren er gratis i alle udbredte Linux-distributioner; det, der koster, er den opmærksomhed, det kræver at holde erklæringen sand. Og en firewall er anden

forsvarslinje. En tjeneste, der kun lytter på loopback-adressen, kan ikke nås udefra, uanset hvad reglerne siger, og det er en stærkere garanti end en afvisningsregel foran en tjeneste, der lytter på alle adresser.

Sådan ser det ud i praksis

Tre stykker af virksomhedens infrastrukturarbejde viser firewallen fra hver sin side: hvordan den kontrolleres, hvornår den oprettes, og hvad der kan gå uden om den.

På én vært stod en hastighedsgrænse for SSH-porten i regellisten i et år, uden at en eneste pakke nåede den, fordi en bredere tilladelse længere oppe matchede først. Statusværktøjerne viste grænsen hele året, og det med rette: den fandtes. Kontrollerne blev skrevet om, så de læser reglernes placering og ikke bare deres tilstedeværelse: de sammenligner nu rækkefølgen i de nummererede regler og i den kæde, kernen faktisk filtrerer med, med det, der var tiltænkt. Hver port får én regel, og hver regel nævner en protokol; SSH er hastighedsbegrænset og har ingen separat tilladelse. De fem offentlige porte er erklæret samlet og kontrolleres som et sæt, så en åben port, erklæringen udelader, gør kontrollen rød, og sikkerhedsrapporten opregner reglerne i den rækkefølge, datapakkerne møder dem, med en note om at læse den rækkefølge.

Den første server blev bygget i hånden og fik først sin firewall senere, hvilket efterlod et tidsrum, hvor en helt ny maskine kunne nås fra internettet. Til server nummer to, hos en ny udbyder, oprettes firewallen af sit eget play, før maskinen findes, så den nye maskine møder internettet med reglerne allerede på plads. Hos den udbyder sættes tags først, fordi en firewall, der rammer tags, kræver, at de findes. Begge udbyderes firewalls er skrevet direkte mod deres REST-API'er gennem en generisk HTTP-opgave, og da den opgave springes over i et prøveløb, øves ændringer på selve firewall-playet, fordi en firewall hos udbyderen koster nul kroner at oprette og kan slettes igen.

Det tredje handler om containere. Docker skriver sine egne poster ind i kernens pakkefilter, og en port, den udgiver for en container, rammes før alt det, værtens hærdningsrolle har lagt ind, så en container kan nås udefra, uanset hvad firewallen har fået besked på. Platformens containere kører under systemd gennem Podman og Quadlet, deler værtens netværk og udgiver ingen porte, så der findes ingen containerport, som noget kan lede uden om reglerne.

Hvor det går galt

En regel står i statusoversigten og rammer aldrig noget. Reglerne læses i rækkefølge, og den første, der matcher, vinder, så en bred tilladelse over en smallere begrænsning eller afvisning skygger den helt. Listen viser stadig begge. Læs reglerne, som kernen gør, nummereret og i rækkefølge, sæt de specifikke regler over de generelle, og test fra en maskine udenfor:

```
ufw status numbered
nft list ruleset
```

En container svarer på en port, firewallen afviser. Docker publicerer porte ved at omdirigere trafikken i kernens NAT-tabel, før den når de kæder, ufw filtrerer, og Dockers egen dokumentation skriver, at firewallens konfiguration dermed reelt ignoreres. Publicér kun til loopback-adressen, sæt en firewall hos udbyderen foran, eller kør containerne uden publicerede porte som beskrevet ovenfor.

IPv4 er sikret, og IPv6 står pivåben. De to adressefamilier har hver sit regelsæt i iptables, og ufw skriver kun IPv6-regler, når IPv6 er slået til i dens standardindstillinger. En vært med en offentlig IPv6-adresse og et regelsæt kun for IPv4 er eksponeret på alle de porte, den lytter på. Brug en nftables-tabel i familien `inet`, som ser begge, eller gennemgå begge lister, og scan værtens IPv6-adresse udefra.

SSH-sessionen fryser i samme øjeblik, firewallen slås til. Default deny trådte i kraft før reglen for SSH, eller SSH-dæmonen blev flyttet til en anden port, mens reglen stadig nævner 22. Tilføj SSH-reglen først, hold udbyderens konsol åben i et andet vindue, og planlæg en automatisk tilbagerulning nogle minutter ude i fremtiden, som annulleres, så snart en ny session er kommet ind.

En regel åbner mere, end forfatteren ville. I ufw åbner en regel, der nævner en port uden protokol, både TCP og UDP. Angiv protokollen i hver regel, og erklær den forventede flade samlet, så alt ekstra får en kontrol til at fejle i stedet for at vente på den næste revision.

Læs mere

- Firewall — de definitioner, NIST's publikationer bruger, hver med henvisning til sin kilde.
- SP 800-41 Rev. 1: Guidelines on Firewalls and Firewall Policy — typer af firewalls, default deny og vedligeholdelse af regler, fra NIST.
- ufw(8) — Ubuntus firewall-frontend, herunder reglernes rækkefølge og reglen `limit`.
- nftables wiki — netfilter-projektets dokumentation af den ramme, der afløser iptables.
- Packet filtering and firewalls — Dockers egen beskrivelse af, hvordan publicerede porte går uden om værtens regler.

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 en firewall?« 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/firewall/