Engineer

Åtkomstkontroll

att bevilja eller neka varje begäran att använda en resurs

Vad det är

Åtkomstkontroll är den del av ett system som besvarar en och samma fråga många gånger i sekunden: får den här anroparen göra det här med den här resursen, just nu? Den kallas också behörighetskontroll, och på engelska access control. Anroparen kan vara en person, en tjänst eller ett schemalagt jobb. Resursen kan vara en fil, en rad i en databas, en API-ändpunkt, ett git-repo eller en dörr. De enskilda mekanismerna som ger svaren heter access controls, och NIST:s ordlista anger authorization, auktorisering, som synonym för hela idén. Definitionerna där täcker båda halvorna av jobbet: beslutet att tillåta eller neka ett subjekt åtkomst till systemets objekt, och processen att bevilja eller neka enskilda begäranden om att hämta och använda information och tillhörande behandlingstjänster, eller om att komma in i bestämda fysiska lokaler.

Den är lätt att blanda ihop med autentisering, men det är två separata steg. NIST SP 800-162 säger det rakt ut: autentisering är inte detsamma som åtkomstkontroll. Autentiseringen fastställer vem som frågar. Åtkomstkontrollen tar den identiteten, tillsammans med allt annat som är känt om begäran, och avgör vad anroparen får göra med den. Ett system kan verifiera en person helt korrekt och ändå låta henne läsa poster som tillhör någon annan, eftersom den andra frågan aldrig ställdes.

Samma vägledning delar arbetet i två. En beslutspunkt (policy decision point) räknar fram svaret, och en verkställighetspunkt (policy enforcement point) sitter i begärans väg och genomför det. Båda kan bo i samma process, men uppdelningen är värd att hålla i huvudet, för ett beslut som ingenting verkställer skyddar ingenting. NIST SP 800-53 skriver in det i två av sina kontroller: AC-24 kräver att beslut tillämpas på varje åtkomstbegäran innan den verkställs, och AC-3 kräver att godkända behörigheter verkställs.

Reglerna kan ta flera former: en lista över tillåtna identiteter på varje objekt, roller som i RBAC, eller attribut hos anroparen, objektet, handlingen och miljön, vilket är ämnet för SP 800-162. Vilken form de än har måste något fortfarande ställa frågan vid varje begäran.

Använd begreppet i ett beslut

Använd ”Vad är åtkomstkontroll?” i ett konkret beslut. Vi kan knyta begreppet till systemet ni bygger eller köper. Konton, lösenord och behörigheter ordnade så att alla har exakt den åtkomst deras arbete kräver, och de som slutar förlorar den samma dag.

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. Skrev 1 000 RBAC-regler som säkrar Linux-servrar — 1,000 åtkomstregler, skrivna för att täppa till en säkerhetslucka

Skrev 1 000 RBAC-regler som säkrar Linux-servrar Hör av er →

Varför det spelar roll

När någon väl är inloggad avgörs nästan allt den personen kan nå av åtkomstkontrollen. Stark autentisering framför svag åtkomstkontroll registrerar bara vilket konto som tog med sig datan. För en verksamhet som förvaltar andras uppgifter är det här som ett löfte om vem som ser vad håller eller brister.

NIST beskriver den ideala verkställighetsmekanismen som en referensmonitor (reference monitor). AC-25 i SP 800-53 kräver en sådan som är manipuleringssäker, som alltid anropas och som är liten nog att analyseras och testas. De tre egenskaperna fungerar som ett praktiskt test för varje behörighetskontroll i en verklig produkt. Kan anroparen ändra de indata kontrollen bygger på, faller den på det första. Finns det en väg som når datan utan att passera kontrollen, faller den på det andra. Är logiken utspridd över hundratals hanterare så att ingen kan läsa den från början till slut, faller den på det tredje.

OWASP Application Security Verification Standard (ASVS) gör om detta till krav som ett team kan testa mot. Kapitlet om auktorisering kräver att åtkomst till funktioner begränsas till anropare med uttrycklig behörighet, att åtkomst till enskilda dataposter begränsas på samma sätt och att reglerna verkställs i ett betrott tjänstelager, i stället för att vila på något som en obetrodd klient kan manipulera, som JavaScript som körs i webbläsaren. Standarden kräver också att en ändring av de värden ett beslut bygger på får effekt direkt. Där det inte går ska en kompenserande åtgärd utlösa ett larm när någon agerar efter att ha förlorat rätten till det.

Avvägningen är en kostnad i anropets väg. Varje beslut behöver fakta, och varje faktum kräver ett uppslag. Den som hoppar över uppslaget brukar flytta kontrollen dit den är billigast, och det är ofta just den plats som anroparen själv styr över.

Så ser det ut i praktiken

Servern har sista ordet om åtkomst i alla tre fallen nedan: på en plattform där en person byter organisation, i ett åtkomstschema skilt från betalningarna och i en git-forge som nekar som standard.

På en sjöfartsplattform där en person kan agera på uppdrag av flera organisationer den aktiva organisationen lagras hos klienten men används inte som betrott underlag för åtkomstbeslut. En OrganizationContext sparar det aktuella valet i localStorage, så bytet sker direkt, och en cookie på serversidan speglar valet så att serverrenderade sidor får samma kontext. Behörigheten kontrolleras på nytt på servern vid varje anrop. Sidan drar gränsen i en enda mening: servern är ensam om att avgöra vad du får göra, och den som manipulerar localStorage ändrar vad det egna gränssnittet visar och ingenting mer, eftersom servern fortfarande säger nej.

På samma slags plattform byggdes betalning och behörighet som två skilda system. Kort via Stripe och köp i appen via Apple och Google strålar samman i ett betalningsschema med 9 tabeller och 34 funktioner. Ett separat åtkomstschema med 11 tabeller och 34 funktioner svarar på ”får det här kontot göra det här?” utan att veta vilken försäljningskanal som betalade. Det svaret styr företagskatalogen, sökningen och dataexporten, och det verifieras på nytt på servern vid varje anrop, eftersom en dold knapp är en artighet och inte ett skydd. Att lägga till en försäljningskanal berör nu bara betalningssidan, och en supportfråga om någons åtkomst har en tabell att slå upp i stället för tre. Sidan anger också priset: två scheman, och ett behörighetsuppslag på anrop som annars hade varit gratis.

När företaget flyttade sin källkod till egen hårdvara sattes git-forgen upp för att neka som standard. Sidan räknar åtkomstkontroll till det man åtar sig när man kör en egen forge. Forgen valdes för att den saknar administrativt webbgränssnitt, med resonemanget att ju mindre gränssnitt, desto mindre att försvara. Grundregeln är att neka: anonyma besökare och den som saknar nyckel släpps inte in, och varje deklarerat repo är uttryckligen privat i stället för att förlita sig på att ingen känner till det. Även vägen dit är avgränsad, eftersom port 23231

bara lyssnar på loopback och den som ska nå den går via en hoppserver. En multiplexer som skulle öppna git över den publika porten är skriven men avslagen, eftersom en lyssnare som ingen använder är attackyta.

Var det går fel

Kontrollen bor i gränssnittet. En knapp döljs eller gråas ut, och ändpunkten bakom den svarar alla som anropar den direkt. Servern måste fatta beslutet på nytt vid varje begäran, utifrån fakta som anroparen själv inte kan ändra.

Funktionen prövas, posten slinker igenom. En begäran pekar ut en faktura, ett dokument eller ett konto med dess identifierare. Servern bekräftar att anroparen får se fakturor och lämnar ut den som efterfrågades utan att kontrollera vems den är. ASVS kallar luckan insecure direct object reference (IDOR) eller broken object level authorization (BOLA). Den som försvarar ett system känner igen den på en identifierare i en begäran som servern använder precis som den kom. Lösningen är att slå upp varje objekt inom anroparens eget behörighetsområde, och i en produkt med flera tenants att ta tenanten från sessionen på serversidan och aldrig från begäran.

Allt nytt står öppet tills någon stänger det. En ny rutt, en ny tabell eller ett nytt repo börjar sitt liv med det som råkar vara standard, och om standarden är att tillåta finns luckan kvar tills någon märker den. SP 800-53 formulerar nätverksvarianten av den motsatta regeln som neka som standard, tillåt som undantag, och samma regel passar på varje lager: det som aldrig har beviljats nekas.

Behörigheterna bara växer. SP 800-162 konstaterar att när åtkomst inte tas bort eller återkallas över tid samlar användare på sig privilegier. Folk byter team, projekt avslutas och inhyrda konsulter går vidare, och vart och ett av de tillfällena är ett ögonblick då något borde tas bort. En återkallelse som får effekt först när en session löper ut flera timmar senare är en mildare variant av samma lucka.

Beslutet lämnar inga spår. När beslut fattas på många ställen och loggas ingenstans står en utredning utan något att läsa. Låt loggningen fånga varje beviljande och varje nekande tillsammans med regeln som gav det, och skydda den loggen: SP 800-162 varnar för att mönstret i besluten kan avslöja attribut hos personerna bakom begärandena.

Läs mer

- Access control, in the NIST CSRC glossary — definitioner från FIPS 201-3 och CNSSI 4009, med authorization angivet som synonym.
- NIST SP 800-53 Rev. 5 — familjen för åtkomstkontroll, bl.a. AC-3 om verkställighet, AC-24 om beslut och AC-25.
- NIST SP 800-162 — attributbaserad åtkomstkontroll, med punkter för beslut och för verkställighet.
- OWASP Application Security Verification Standard — testbara krav på auktorisering för funktioner, data och enskilda fält.

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 åtkomstkontroll?” 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/access-control/