Liikenne- ja viestintävirasto
Transport- och kommunikationsverket

Huoltovarmuuskeskus
Försörjningsberedskapscentralen
National Emergency Supply Agency

Cybersäkerhet för AI-agenter

Anvisningar för säker planering och införande av AI-agenter
Sammanfattning

Traficom
Försörjningsberedskapscentralen

Traficoms publikationer
5/2026

Innehållsförteckning

1 Inledning och mål

AI-agenter blir snabbt vanligare och möjliggör en mångsidig automatisering av uppgifter – från informationssökning till programmering – men medför betydande informationssäker-hetsrisker som inte alltid beaktas i tillräcklig grad när de tas i bruk. En agent lämpar sig inte heller för allt: för enklare analyser och klassificeringar är traditionella maskininlär-ningsmetoder ofta ett bättre val och det lönar sig att överväga agenter endast när pro-blemet är tillräckligt komplext och inte kan lösas med en enskild metod.

Syftet med denna anvisning är att hjälpa organisationer att planera, bygga och driva agentiska AI-system på ett säkert sätt. Utredningen stöder sig på den i skrivande stund senaste tillgängliga informationen om de vanligaste informationssäkerhetsriskerna i an-slutning till AI-agenter och bästa praxis inom hotmodellering. Med hjälp av dessa kan varje organisation börja planera och bygga AI-agenter utan att äventyra informationssä-kerheten oskäligt. Anvisningarna är särskilt avsedda för personer som arbetar med artifici-ell intelligens och informations- och cybersäkerhet. Anvisningarna har skapats i samarbete mellan Traficom och Försörjningsberedskapscentralen inom ramen för programmet Digi-taalinen turvallisuus 2030 (Digital säkerhet 2030) och som en del av beredskapen inför framtiden. Utredningen har skrivits i samarbete med Solita. Flera företag och andra intres-sentgrupper har också deltagit i utarbetandet av utredningen.

2 AI-agenter och deras särdrag

En AI-agent är ett informationssystem som utnyttjar AI-modeller och olika verktyg för att utföra funktioner automatiskt. Till skillnad från AI-assistenter kan agenter automatiskt kedja samman åtgärder utan kontinuerlig mänsklig styrning.

2.1 Tre huvudkomponenter

- **Basmodell:** den bakomliggande språkmodellen (LLM) som bearbetar naturligt språk
- **Promptar:** en systemprompt styr agentens verksamhet; även användarens indata i modellen kallas prompt
- **Verktyg:** funktioner eller gränssnitt för datasökning (t.ex. informationssökning på nätet), åtgärder (t.ex. produktion av tabeller eller grafiska representanter) och orkestrering (assistera eller anropa andra agenter)

2.2 Agenttyper

Det finns flera olika typer av agenter och sätt att kategorisera dem. Det finns listor till exempel på IBM:s och Red Hats webbplatser[1]. De vanligaste agenterna är:

Typ	Beskrivning	Exempel
• Enkel agent	• En regelbaserad agent som utför en specifik uppgift.	• Agenten behandlar en mottagen fil och lägger till metadata enligt instruktioner
• Lärande agent	• Utnyttjar tidigare respons för att förbättra sin funktion	• Agenten sammanställer en ticket i felsituationer eller rekommenderar åtgärder i avvikande situationer. Det är alltid en människa som ger agenten respons om kvaliteten på dess verksamhet och med hjälp av responsen kan agenten förbättra sin funktion.
• Multiagent system	• Flera agenter som samarbetar, mer komplexa uppgifter	• Ett centraliserat system där en samordnande agent tar emot uppgifter och delar ut dem till agenter som är specialiserade på specifika uppgifter.

[1] IBM: https://www.ibm.com/think/topics/ai-agent-types
Red Hat: https://www.redhat.com/en/blog/understanding-ai-agent-types-simple-complex

3 Centrala sårbarheter i informationssäkerheten

Guiden stöder sig på de risker OWASP identifierat i anknytning till AI-agenter och LLM-system. I december 2025 publicerade OWASP en separat förteckning över riskerna för-knippade med agentiska system (ASI01–ASI10).[2]

Typiska hot specifika för agenter (OWASP ASI Top 10)

Kod	Hot	Beskrivning
ASI01	**Manipulation av mål**	Med hjälp av promptinjektioner redigeras agentens mål och beslutsfattande – agenten skiljer inte mellan instruktion-erna och angriparens input.
ASI02	**Missbruk av verktyg**	Tillåtna verktyg missbrukas: dataläckage, manipulation av utdata, utvidgning av användarbehörigheter.
ASI03	**Identitets- och behörighetsma-nipulation**	Manipulation av delegeringskedjor gör det möjligt att göra behörighetseskale-ring och förbigå övervakningen.
ASI04	**Leverantörske-djerisker**	Tredjepartskomponenter (modeller, verk-tyg, data) kan innehålla skadlig kod eller dolda instruktioner.
ASI05	**Fjärrexekvering av kod**	Kodskapande egenskaper utnyttjas för att utvidga en attack till att fjärrexekvera kod (RCE).
ASI06	**Minnes-manipulering**	Genom att förgifta RAG-datalager eller agentens kontext förvränger man slutled-ningen och orsakar dataläckage.
ASI07	**Osäker kommu-nikation**	Svag autentisering mellan agenter möj-liggör kapning, manipulation och förfalsk-ning.

[2] [2]OWASP TOP 10 For Agentic Applications 2026: https://genai.owasp.org/resource/owasp-top-10-for-agentic-applications-for-2026/
OWASP top 10 for LLM applications: https://owasp.org/www-project-top-10-for-large-language-model-applications/

ASI08	**Kedjereaktioner**	Ett enskilt fel eller en skadlig indata kan spridas till hela multiagentsystemet.
ASI09	**Utnyttjande av förtroende**	Människors överdriven tillit till agentens rekommendationer leder till skadliga beslut och dataläckage.
ASI10	**Okontrollerade agenter**	Agenten följer inte den planerade funktionen och saboterar systemen externt samtidigt som den verkar fungera normalt.

Hot förknippade med LLM-system (OWASP LLM Top 10)

Kod	Hot	Beskrivning
LLM01	**Promptinjektion**	Skadliga promptar formar modellens funktion – svårt att upptäcka, kan också inriktas indirekt via externa källor.
LLM02	**Känslig information**	Språkmodellen avslöjar konfidentiell information via träningsdata, användarindata eller databasåtkomsten.
LLM03	**Leveranskedja**	Tredjepartskomponenter kan innehålla bakdörrar; öppna modeller (LoRA, PEFT) ökar risken.
LLM04	**Förgiftning av data**	Manipulation av träningsdata eller kontexten äventyrar modellens integritet – aktiveras eventuellt först av en viss utlösare.
LLM05	**Felaktigt utdata**	En ovaliderat LLM-utdata möjliggör XSS-, CSRF- och SQL-injektionsattacker eller fjärrexekvering av kod.
LLM06	**Alltför omfattande behörighet**	Alltför omfattande behörigheter eller autonomi gör det möjligt att kedja

		samman skadliga funktioner utan övervakning.
LLM07	**Promptläckage**	Röjande av en systemprompt kan röja skydd och hjälpa angriparen att kringgå dem.
LLM08	**Vektorsvagheter**	RAG-systemens vektordatabaser ökar risken för skadligt innehåll och förgiftning av minnet.
LLM09	**Felaktig information**	Hallucinationer och felaktig information som produceras av AI kan leda till allvarliga beslut och anseendeskador.
LLM10	**Obegränsad förbrukning**	Obegränsade förfrågningar belastar systemet, orsakar tjänsteavbrott och kostnadsrisker.

4 Säker designav AI-agenter

Identifieringen och hanteringen av riskerna börjar i planeringsskedet. Före de tekniska valen måste man svara på tre grundläggande frågor:

- Vilken process blir agenten en del av och vilken är målbilden?
- Vilken är den positiva effekten som eftersträvas med AI och för vem?
- Vilka centrala begränsningar gäller för processen?

4.1 Preliminär riskbedömning – fyra perspektiv

Användningsfall

- Ett avgränsat användningsfall är säkrare: Ju bredare användningsfallet är, desto lättare är det att påverka agentens funktion med illvilja. För att avgränsa an-vändningsfallet kan agentens funktion begränsas så mycket som möjligt med systempromptar och andra tekniska metoder. Om användningsfallet är omfat-tande finns det färre begränsningar och det uppstår luckor betydligt lättare
- Beakta sannolikheten för att användningsfallet förändras: Om användarna har många olika användningsfall för vilka de vill använda AI, men endast tillgång till ett fåtal säkra verktyg som godkänns av organisationen, kan det förekomma press på att använda en agent med begränsat användningsändamål även för andra ändamål. Detta kan öka risken för till exempel missbruk av agentensut-data, alltför omfattande befogenheter eller röjande av känslig information.
- Planera in monitoreringen av implementeringensom en del arkitekturen från början: Monitorering av användningsfallet är inte bara ett tekniskt problem, ef-tersom det ökar monitoreringen av den anställdas arbetsuppgifter i detalj. Därför är det skäl att redan i planeringsskedet fundera på hurdan monitorering som kan genomföras för att man på ett meningsfullt sätt ska kunna följa upp hur an-vändningsändamålet förändras och hur ändamålsenligt det är och upptäcka sår-barheter.
- Bedöm användargruppens förmåga att upptäcka fel och felaktig information: An-vändarens agerande är en av nyckelpunkterna i informationssäkerheten. Därför är det viktigt att redan i planeringsskedet skapa sig en uppfattning av hur san-nolikt det är att användarna förstår i vilka fall agenterna är tillförlitliga och för vilka ändamål det inte lönar sig att använda agenter.

Datalager

- Ju känsligare data agenten använder, desto mer riskfylld är lösningen
- Minimera data: ge agenten endast den information som är väsentlig för uppgif-ten
- Identifiera motstridigheter i fråga om användarbehörigheter redan i planerings-skedet

Integrationer

- Identifiera hur kritiska system agenten behöver åtkomst till
- Bedöm om åtkomsten till kritiska system kan begränsas på ett meningsfullt sätt

Användarnas delaktighet

- Intervjua användare för att kartlägga informationssäkerhetsrisker
- Samplanering avslöjar risker tidigt
- Testa de första versionerna tillsammans med användarna

Förordningen om artificiell intelligens ((EU) 2024/1689) delar in systemen för artificiell intelligens i fyra riskklasser: förbjudna användningsområden, system med hög risk, system som omfattas av transparensskyldigheterna samt andra system. Varje leverantör ska säkerställa att sitt system är lag-enligt.

	Låg	Måttlig	Hög	Mycket hög
Inverkan på kärnverksamheten	Systemet utför inte uppgifter som hänför sig till kärnverksamheten.	Systemet utför lätt hanterbara och begränsade delar av arbetsflöde som är en del av kärnverksamheten.	Systemet utför en del av organisationens kärnverksamhet.	Systemet utför centrala delar av organisationens kärnverksamhet.
Autonomi och komplexitet	Systemet är inte integrerat med andra system eller andra agenter.	Systemet är integrerat med ett litet antal externa system som inte är kritiska för kärnverksamheten.	Systemet är integrerat med flera externa verktyg och/eller utgör ett multiagentsystem. En del av integrationerna hänför sig till system som är kritiska för kärnverksamheten.	Systemet är ett fleragentsystem med flera gränssnitt och integrationer med system som är kritiska för kärnverksamheten.
Datans känslighet	Systemet behandlar endast offentliga uppgifter.	Systemet behandlar organisationens interna information.	Systemet behandlar konfidentiell information.	Systemet behandlar sekretessbelagd och känslig information.

Figur 3. Matris till stöd för riskbedömning

I riskbedömningen ska man beakta åtminstone hur kritiskt systemet är för kärnaffärsverk-samheten, den nivå av autonomi och komplexitet som lösningen kräver samt känsligheten hos de data som agenten behandlar.

5 Hotmodellering

Hotmodellering är ett strukturerat tillvägagångssätt för att identifiera informationssäker-hetsrisker. Den ska inledas i början av projektet och upprepas alltid i samband med bety-dande förändringar. För AI-agenter rekommenderas referensramen MAESTRO vid sidan av traditionella metoder (STRIDE, LINDDUN).

5.1 Tre faser

Fas 1: Definition av objektet: Identifiera information, användare och använd-arbe-höriigheter, beroenden och informationsflöden som ska skyddas. Utarbeta ett data-flödesschema med tillförlitlighetsgränser.

Fas 2: Identifiering av hot: Kartlägg hotaktörer, motivationer och attackvekto-rer. Utnyttja OWASP Top 10-listorna, MITRE ATLAS-databasen och information om tradit-ionella cyberhot.

Fas 3: Bedömning av hot och kontrollåtgärder: Hoten prioriteras på basis av sanno-likhet och verkningar. Kontrollåtgärderna enligt referensramen NIST de-finieras: skydd, observation, åtgärd, återställning.

Hotmodelleringen genomförs i praktiken i workshoppar (2–3 timmar/fas). Det rekommen-deras att affärsägare, produktägare, arkitekter och datasäkerhets-ansvariga deltar.

Ett sätt att strukturera hot i anslutning till agentlösningar är metoden MAESTRO, Multi-Agent Environment, Security, Threat, Risk and Outcome. Den är avsedd att komplettera traditionella metoder med ett tillvägagångssätt i flera lager när man skapar hotmodeller för AI-lösningar.

Hotmodellen är en kontinuerlig, levande dokumentation – inte en engångspro-cess.

Exempel: Som slutresultat av hotmodelleringen skapas en hotmodell för orga-nisationen som innehåller identifierade hot samt bedömningar av deras sanno-likhet och verkningar. Det fastställs ändamålsenliga kontrollåtgärder mot hoten, och det finns planer för meto-dernas genomförande och tidtabell. I utveckl-ingen av agentlösningen beaktas de hot som är typiska för lösningen i fråga. Kontrollåtgärdertas fram mot hoten, till exempel begräns-ning av antalet för-frågningar och filtrering av indata och utdata. Ansvarspersonen för in imple-mentering av kontrollåtgärdernai utvecklingsarbetets arbetsköer och ser till att hot-modellen uppdateras när kontrollåtgärderna har tagits i bruk. En genom-gång av hotmo-dellen schemaläggs enligt överenskommelse för följande kvar-tal, om det inte i utveckl-ingsarbetet framkommer behov av att gå igenom hot-modellen tidigare. Det rekommende-ras att man fastställer vilka förändringar som är så betydande att de kräver en omprövning av hotmodellen. Sådana

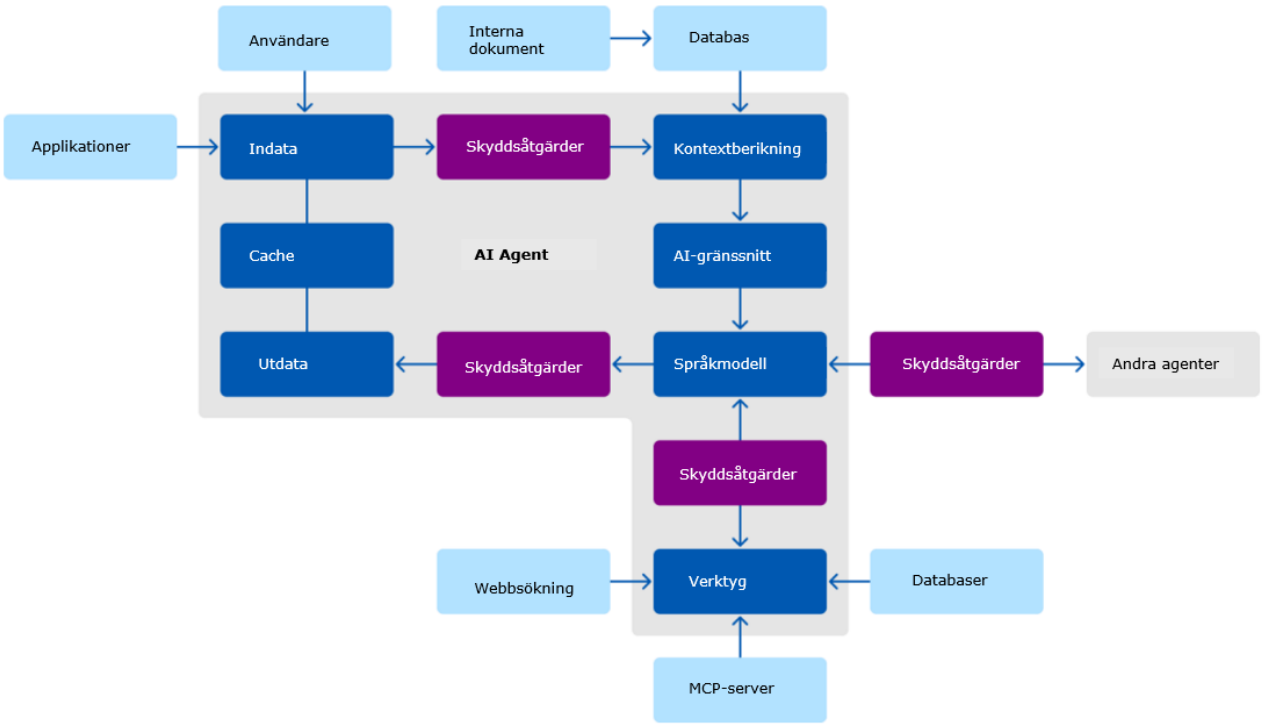
förändringar kan till exempel vara uppdatering av modellen, en ny modell, en ny integration, ett nytt användningsfall eller ändringar i systemets infrastruktur.

Byggande av AI-agenter

När hoten har modellerats, kontrollåtgärderna fastställts och ansvaret fördelats kan man gå vidare till att bygga en AI-agent. I detta avsnitt beskrivs hur olika val i byggskedet påverkar agentens säkerhet med hänsyn till de allmänt kända riskerna. I avsnittet besk-rivs val som hänför sig till valet, behandlingen och förvaringen av data, anmärkningar i anslutning till stora språkmodeller och frågor gällande operationen av agenten. Med hjälp av dessa anvisningar kan organisationen bygga sig en agent enligt dagens bästa säker-hetspraxis.

5.2 Data och datahantering

- Bedöm datakvaliteten, datasäkerheten och deras värde före använd-ning: bra data är en förutsättning för agentens funktion
- Beakta eventuella hallucinationer och snedvridningar i träningsdata (ålder, kön, nat-ionalitet)
- Versionera data för att möjliggöra spårbarhet; förvara testdata sepa-rat
- Säkerställ att data är aktuella: särskilt viktigt i RAG-arkitekturer
- Följ principerna för minimering av data och nolltillit vid integrationer



Figur 4. AI-agentens komponenter

5.3 Säker promptutformning

Språkmodellernas utdata grundar sig alltid på en prompt. En prompt består av flera olika komponenter, och flera språkmodellsleverantörer har gränssnitt som gör det möjligt att dela upp prompten i logiska komponenter:

- **Systemprompt:** Fastställer funktionsprinciperna för agentens roll, kontext och språkmodell i olika situationer.
- **Användarinmatning:** Berättar vad språkmodellen ska göra. Indata kan till exempel vara en fråga i en chattapplikation eller en definition av en uppgift som inleder pro-cessautomationen.
- **Berikning av promptar:** Språkmodellens utdata kan ofta förbättras genom att ge modellen kontext med hjälp av externa dokument. Språkmodellens svar kan förbätt-ras till exempel genom att dela den information som användaren har med språkmo-dellen.
- **Agentens utdata:** Slutlig utdata kan till exempel vara ett meddelande som returne-ras av ett externt system, ett resultat som beräknaren returnerat eller en applikation som språkmodellen byggt. Dessa resultatkan användas för att automatiskt starta nya uppgifter.
- **Historik:** Det är möjligt att mata in historik av tidigare indata och utdata i språk-modellen och med hjälp av dessa kan agentens funktioner förenhetligas. När prompt-en utarbetas är det viktigt att beakta att varje komponent är likvärdig för språkmo-dellen. Detta innebär att alla delar av en prompt kan styramodellens beteende och göra det möjligt för till exempel angripare att injicera en prompt.

Det är alltså viktigt att skydda prompten mot hot. OWASP:s anvisning för skydd av promptar:

1. Begränsa modellens funktion. Instruera agenten att inte beakta andra instruktioner. Påminn modellen efter att den läst externa dokument om dess uppgift så att eventuella instruktioner från angripare förbigås. Dessutom kan man instruera språkmodellen att reagera endast på en viss typ av indata.

2. Validera formatet för språkmodellens output. Det format som definierats för språkmodellens utdata kan valideras med ett annat program.

3. Bygg skyddsåtgärder för promptarna och utdata. Kontrollera till exempel med en annan språkmodell att prompten inte innehåller otillåtna ämnen och att prompten överensstämmer med agentens uppgift. Det är viktigt att notera att användningen av en annan språkmodell för kontroll inte är en felfri skyddsåtgärd. Det rekommenderas att andra metoder används då det är möjligt.

4. Ge språkmodellen minimala behörigheter. Begränsa språkmodellens behörigheter så att endast nödvändiga uppgifter kan genomföras. Genom att begränsa läsbehörig-heterna till filer undviker man till exempel att känslig information läcker ut.

5. Förutsätt att en människa måste godkänna kritiska beslut. Förutsätt att kritiska operationer kräver bekräftelse av användaren. Bekräftelse minskar risken för olovliga operationer.

6. Separera okänt innehåll tydligt från den övriga prompten. Med okänt innehåll av-ses externa dokument, resultat av sökningar på internet samt användarens prompt. Den ska separeras tydligt från den övriga prompten så att dokumentens innehåll inte påverkar språkmodellens funktion. Okänt innehåll kan specificeras till exempel med de separators som nämns i anvisningarna för en bra prompt.

7. Testa aktivt modellens funktion. Bygg upp tester eller simuleringar med syfte att bryta modellens skydd. Syftet med testerna är att utreda modellens gränser och den verkliga omfattningen av åtkomsträttigheterna.

5.4 Skyddsåtgärder (Guardrails)

Skyddsåtgärderna säkerställer att språkmodellen fungerar som planerat. De genomförs både för indata och utdata: identifiering av teckensträngar, jämförelse mot förbjudna ord, klassificering av indata, identifiering av personuppgifter och kontroll av hallucinationer. Specialiserade modeller är bl.a. Llama Guard och Constitutional Classifiers.

Skyddsåtgärderna bildar en grupp regler och anvisningar som säkerställer att språkmo-dellen fungerar som planerat. För varje användningsfall ska man skapa egna regler för hur modellen ska bete sig i olika situationer. Skyddsåtgärderna kan genomföras både för in-data och för den information som språkmodellen returnerar till exempel med följande me-toder:

1. Identifiering av teckensträngar i texten: Identifiera ord eller teckensträngar i texten som språkmodellen inte ska behandla eller generera för användaren. Detta är till ex-empel personuppgifter, såsom personbeteckningar eller e-postadresser

2. Jämförelse av indata mot förbjudna ord: Det är möjligt att på förhand definiera ord/teman som en AI-agent inte får behandla.

3. Klassificering av indata i tillåtna kategorier med en separat modell: Texten kan med en separat modell klassificeras i olika kategorier som agenten får ge svar på.

Indatasäkerhet kan förbättras genom att identifiera till exempel följande information:

4. Farligt eller skadligt innehåll: Identifiering av farlig eller skadlig indata, med vilken man till exempel försöker hitta sårbarheter i språkmodellen, stjäla en systemprompt eller skapa skadligt innehåll.

5. Personuppgifter: Innan en språkmodell anropas kan till exempel personuppgifterna i indata döljas. Om personuppgifterna behövs i det slutliga svaret kan de läggas till i den text som modellen skapar.

En till synes harmlös indata kan eventuellt få modellen att återställa information som man inte vill att ska vara synlig för användaren. Vid granskningen av det returnerade värdet kan man beakta till exempel:

6. Innehåll, personuppgifter eller skadlig information: Skyddsåtgärderna ska implementeras så att innehållet motsvarar användarens förfrågan och så att det inte innehåller innehåll som den som utvecklar språkmodellen fastställt som skadligt, t.ex. desinformation.

7. Hallucination: Hallucination är ett vanligt problem hos språkmodeller. Slutresultatet kan till exempel jämföras mot indata och externa dokument för att säkerställa att information-en är korrekt.

8. Syntax: Språkmodellens slutresultat kan valideras så att det till exempel motsvarar ett visst programmeringsspråk. Indata kan kontrolleras så att den till exempel överensstäm-mer med JSON-syntaxen.

5.5 Verktyg och agenternas autonomi

För att agenten ska kunna använda verktyg på ett säkert sätt måste man i valet av verk-tyg säkerställa vad verktyget gör, vilken information som krävs för att använda det och var verktygets beräkning sker. Nedan listas exempel på åtgärder för att hantera riskerna med verktyg som används av AI-agenter.

1. Körning av program: Innehållet i en applikation som genererats av en AI-agent ska kontrolleras och applikationen ska köras i en begränsad miljö.

2. Begränsning av användarbehörigheter: Användarbehörigheterna till verktygen ska be-gränsas så att de är så snäva som möjligt. Till exempel ska man endast ge begränsade behörigheter till databaser så att agenten endast kan söka eller skriva nödvändig inform-ation i databasen.

3. Ersättande av verifierat verktyg med ett skadligt: En del av verktygen kan köras på externa servrar (se t.ex. MCP-server). Då kan innehållet i ett tidigare säkert verktyg göras skadligt. Det kan till exempel läggas till nya funktioner i verktyget som skickar de uppgif-ter som verktyget behandlar per e-post till den som modifierat verktyget. När verktyget används kan man till exempel kontrollera verktygets hashvärde.

En självständig agent är sårbar för manipulering av målet. En angripare kan med hjälp av en promptinjektion försöka ändra agentens planering eller slutledning för att utföra skad-liga åtgärder, såsom nätfiske av kunduppgifter i systemet. Följande åtgärder är centrala för att skydda agentsystemets uppgiftskedjor:

- Begränsa användarbehörigheterna så att de är så snäva som möjligt så att agenten inte kan vidta åtgärder utanför den angivna uppgiften
- Begränsa tillgången till verktyg till endast sådana verktyg som agenten behöver för att utföra uppgiften
- Övervaka AI-agenter för att identifiera mål och beteendeförändringar.
- Sätt gränser för hur många olika agenter som kan anropas

6 Testning och monitorering

AI-agenter ska testas regelbundet – att testa en gång för godkännande räcker inte. AI:s natur (indeterminism, kontinuerligt lärande) ställer särskilda krav på testningsstrategierna.

6.1 Testningens nivåer

- **Applikationsnivå**: promptinjektioner, läckage av känsliga data, röjande av systemprompt
- **Modellnivå**: hållbarhet, sårbarheter, förgiftning, integritet
- **Infrastrukturnivå**: leveranskedjor, resurser, säkra konfigurationer
- **Datanivå:** datakvalitet, integritet, skadligt innehåll, röjande av träningsdata

6.2 Testtyper

- **Hybridtestning:** kombinerar automatiserad testning och expertbedömning; metoden LLM-as-a-judge
- **Adversial testning**: skadligt innehåll, stresstestning, red teaming (Garak, LLAMATOR, PyRIT)
- **Simuleringstestning:** imitation av dynamiska interaktioner mellan användare och agenter (τ-bench)
- **Regressionstestning:** automatiserat särskilt i samband med uppdatering av modellen

6.3 Monitorering och hantering av incidenter

Kontinuerlig monitorering är nödvändig: man följer upp beslutsfattandet, prestanda och interaktionen. Incidenter (missbruk, promptattacker, hallucinerande agenter) upptäcks genom att jämföra resultaten med valideringens basnivå. I synnerhet i multiagentsystem gör loggen det möjligt att spåra fel till en enskild komponent.

AI-agenterna lär sig och anpassar sig i takt med att data och användare förändras, så kontinuerlig monitorering är nödvändig för att säkerställa att de fungerar i överensstämmelse med sitt användningsändamål. Agenternas funktion utvärderas genom att jämföra resultaten med valideringens basnivåer och testernas prestanda följs upp för att identifiera felaktiga verksamhetsmodeller.

Eftersom agentens status inte är bestående utan finjusteras kontinuerligt, måste även testfallen uppdateras så att de motsvarar agentens nuvarande status. Beteende och resultat ska dokumenteras för att även långsiktiga förändringar ska kunna upptäckas.

När det gäller agenter inom produktionen ska funktionen, beslutsfattandet, prestanda samt interaktionen med användare och externa aktörer övervakas. För att upptäcka missbruk ska särskilt användningsmängder, anropade verktyg, agentens uppgifter och skyddsåtgärder följas upp. Monitoreringen avslöjar

också inkonsekvenser i besluten – till exempel ett överraskande godkännande av tidigare förkastade åtgärder kan tyda på ett angrepp.

Incidenterna kan bero på systemfel, förvrängda algoritmer eller skadlig verksamhet, såsom promptangrepp. De måste upptäckas snabbt för att förhindra felaktig eller skadlig funktion. För detta utnyttjas automatisk övervakning, algoritmer som lämpar sig för identifiering av incidenter samt användarrapportering. I ett multiagentsystem hjälper monitoreringen med att spåra fel till en viss komponent. Även om utredningen ofta sker manuellt stöder automatiseringen datainsamling och analys.

I händelse av incidenter kan det behövas omedelbara begränsande åtgärder, såsom att koppla bort systemet från nätet eller förhindra användningen av känslig data. Korrigerande åtgärder är till exempel uppdatering av programvara och algoritmer, avbrytande av användningen eller användning av alternativa lösningar. Det är viktigt att förutse incidenter, utarbeta beredskapsåtgärder och upprätthålla en kontinuerlig monitorering. Incidenthanteringen ska skräddarsys enligt varje organisations system och hotscenarier.

7 Upphandlingsanvisning för agentiska system

I denna bilaga erbjuder vi en minneslista för upphandling av AI-agenter som köparen bör kontrollera med agentens leverantör. Det finns många kriterier och det är inte alltid nödvändigt att anta att varje system ska uppfylla alla av dem. Det lönar sig alltså att plocka ut de omständigheter ur listan som är viktigast med tanke på användningsområdet, uppgiften, tryggandet av kärnverksamheten och organisationens mål.

Syftet med anvisningarna är att komplettera organisationernas befintliga säkerhetsanvisningar och säkerhetspolicyer. Det är viktigt att de AI-agenter som upphandlas också utvärderas i enlighet med de befintliga anvisningarna om programvarusäkerhet.

Hantering och livscykelunderhåll

Kontrollera vilka språkmodeller leverantören erbjuder och att de passar användningsfallet.

- Det finns språkmodeller från flera olika leverantörer. Hur omfattande är tjänsteleverantörens utbud? Hurdan publiceringsfördröjning har leverantören på modellerna?

☐ Kontrollera i vilken grad leverantören tillåter export av agenter till andra system.

- Prompten, egna data och specifikationer är en viktig del av utvecklingen av AI-agenter. Genom att säkerställa hur uppgifterna hämtas från applikationen är det enklare att övergå till en ny tjänsteleverantör vid behov.

☐ Säkerställ att det system som upphandlas möjliggör livscykelhantering när språkmodellerna uppdateras.

- Kan samma språkmodell användas åtminstone över organisationens egen publikationscykel eller flera cykler? Finns det tid att testa det som är nytt när nuvarande system åldras?
- Vilken uppdateringsväg erbjuder leverantören för nya språkmodeller? Vilka av den nya modellens funktioner eller förmågor kan tas i bruk direkt, och vilka kräver ändringar i den applikation, det system eller den agent som använder modellerna?
- Kan man återgå till en tidigare modell eller publicering om så behövs?
- Hur garanterar man att språkmodellen är tillgänglig inom ett specifikt geografiskt område?

☐ Se också till att leveransen innehåller

- ModelOps-/agentversionshantering.
- CI/CD-pipelines för säker driftsättning och återställning.
- administrationspaneler för uppföljning av systemets status och justering.
- möjlighet att tvinga att administrationsregler tillämpas (t.ex. eskaleringsgränser).

- dokumentation av leverantörens uppdaterings- och korrigeringspolicy.
- bevis på kontinuerligt deltagande i forskning, utveckling och standardisering.

Funktionalitet och integrationer

☐ Kontrollera att applikationen erbjuder möjlighet att spara egna dokument.

☐ Kontrollera att applikationen erbjuder en lämplig arkitektur för bearbetning av uppgifter.

- Stöder applikationen RAG-arkitektur?
- Vilka inbäddningsmodeller används?

☐ Se till att agenten kan anpassas vid behov.

- Hur kan man redigera agenter?
- Är det enkelt att definiera promptar, verktyg och uppgifter?

☐ Kontrollera om genomförandet omfattar en färdig tjänst för testning av agenter i flera olika miljöer.

☐ Kontrollera att agenten är korrekt övervakad.

- Hur sköter tjänsteleverantören monitoreringen?
- Kan man redigera de komponenter som ska monitoreras?
- Kan man utöver monitoreringen skicka larm/anmälningar exempelvis per e-post?

☐ Kontrollera om tjänsten stöder multiagentsystem särskilt om användningsfallet kräver det nu eller eventuellt i framtiden.

☐ Se till att det är möjligt att ansluta nödvändiga verktyg till agenten.

- Är det möjligt att lägga till egna verktyg för AI-agenter i systemet?
- Stöder leverantören det senaste protokollen (t.ex. MCP, A2A)?
- Hur lägger man till nya verktyg i systemet?
- Vem ”äger” de nya verktygen?

☐ Kontrollera att leverantören har färdiga gränssnitt (API) eller gränssnitt som kan redigeras även med tanke på företagets systemintegration.

- Kan systemet integreras med de cybersäkerhetslösningar som din organisation använder (t.ex. SIEM)?

☐ Kontrollera att leverantören har säkerställt stöd för standarder (t.ex. FIPA ACL, RESTful API).

☐ Se till att agenten vid behov också kan integreras med de gamla systemen.

Säkerhet

Se till att användaradministrationen är säker.

- Stöder applikationen användaradministration?
- Hur säkerställs åtkomsträttigheterna till olika data?
- Kan användningen av systemet begränsas med en IP-adress?
- Bekräftelse av agenternas identitet (IAM-integration, RBAC).

☐ Säkerställ att hanteringen av verktyg och integrationer samt deras säkerhet har beaktats då AI-agenter använder verktygen. Utan exakt validering av verktygen kan agenterna utföra oönskade uppgifter, såsom att skicka företagets data till externa system.

- Var körs verktygen?
- Hur säkerställs verktygens informationssäkerhet?
- Är det möjligt att kräva att en människa bekräftar varje verktygsanrop?
- Är verktygens beskrivningar och funktioner tillgängliga för användarna?
- Kan AI-agenten kopplas till verktyg utanför applikationen (t.ex. egna MCP-servrar).

☐ Se till att agenten har tillräckliga skyddsåtgärder för att validera indata och det värde som returneras.

- Omfattar applikationen kontroller av indata och det värde som returneras?
- Kan man bygga upp egna valideringar i systemet?

☐ Säkerställ dessutom följande skyddsåtgärder:

- Zero trust-kommunikation (ömsesidig identifiering, TLS-kryptering).
- Säkerhetsgateway/sandbox för externa verktygs- och API-anrop.
- Leverantören har tillräcklig hotmodelleringspraxis till exempel för promptinjektioner, identitetsförfalskningar och förgiftning av data.
- Monitorering och upptäckt av incidenter i fråga om agenternas misstänkta beteende.

☐ Om du vill kan du också kontrollera om leverantören stöder poängsättning av tillit/anseende för agenter

Dataskydd och reglering

Se till att leverantören har beaktat åtminstone följande i fråga om dataskyddet:

- Minimering och klassificering av data mellan agenter
- Kryptering i vila och under överföring.
- GDPR/HIPAA (och annan reglering som motsvarar organisationen och användningsändamålet) så långt som möjligt inbyggd.
- Lagring och raderingspolicy kan fastställas
- Leverantören följer de referensramar som krävs (t.ex. NIST AI RMF / ISO 42001 / AI TRiSM)

Transparens och revisionsbarhet

☐ Se till att agenten har en global revisionslogg över agenternas alla aktiviteter och interaktioner.

☐ Kontrollera att agentlösningen erbjuder tillräckliga verktyg för att garantera förklarbarhet (beslutskedjor, förklaringsloggar).

☐ Säkerställ att systemet stöder möjligheten för mänsklig övervakning och styrning i realtid.

☐ Kontrollera att leverantören kan påvisa samarbetsindikatorer (t.ex. Component Synergy Score).

Övrigt

☐ Säkerställ följande i fråga om fakturering:

- Hur fungerar faktureringen för applikationen? AI-agenter tar upp mycket beräkningseffekt, så kontrollera vad som faktureras och hur.
- Hur förutsägbara är kostnaderna? Förändringar och variationer i agentens användning påverkar kostnadsfördelningen och förutsägbarheten.

☐ Kontrollera också om din organisation har andra hanteringsprocesser (för artificiell intelligens) som just din organisation måste beakta för att den agentlösning som skaffats inte ska äventyra kärnverksamheten. Exempel på branschspecifika frågor som ska beaktas:

- Säkerhetsreglering av kritisk infrastruktur
- Etiska principer för vårdarbete och reglering av medicintekniska produkter
- Transparens och lagenlighet som krävs av demokratiska processer (inkl. förvaltningslagens krav på automatiserat beslutsfattande)

Transport- och kommunikationsverket Traficom
PB 320, 00059 TRAFICOM
tfn 029 534 5000

traficom.fi

ISBN 987-952-425-008-5
ISSN 2669-8787 (elektronisk publikation)

TRAFICOM
Liikenne- ja viestintävirasto