

MODULO 14

Costruire l'AI difensiva e AI forensics

Progettare un assistente di triage con RAG, valutarlo prima del rilascio e renderlo indagabile, considerando il sistema AI come possibile oggetto di indagine forense.

RAG

Ollama / vLLM

AI FORENSICS

Abstract e obiettivi formativi

Abstract

Il modulo guida la progettazione di componenti AI per la difesa, come architetture RAG su playbook e threat intelligence eseguite in locale e valutate sistematicamente prima del rilascio, e introduce l'analisi forense dei sistemi AI: quali eventi registrare e come ricostruire, dall'audit trail, la catena causale di una decisione errata o di un abuso.

Obiettivi formativi (Bloom)

- **Creare:** un prototipo di assistente di triage con RAG su documentazione interna
- **Valutare:** metriche prima del rilascio (groundedness, tasso di errore, latenza)
- **Analizzare:** un incidente su un'applicazione LLM a partire dal suo audit trail

Traccia core

01 02 07

Detection & Response

03 04 05 06

Digital Forensics

08 09 10 11

Sicurezza dell'AI

06 12 13 14

Traccia core

15 16

Letture di raccordo: § 10.8

Ciclo di vita di un assistente AI difensivo



Assistente di triage basato su RAG



RAG e fine-tuning a confronto

Nel SOC il RAG è in genere preferibile: la conoscenza cambia con frequenza, le fonti devono essere citabili e l'aggiornamento richiede una nuova indicizzazione anziché un nuovo addestramento. Il fine-tuning è adatto a stile e formato dell'output più che all'aggiunta di conoscenze. Il server di inferenza richiede autenticazione, modelli di provenienza verificata e limiti al consumo di risorse, come ogni servizio critico.

Valutazione prima del rilascio

GROUNDNESS

Quota di affermazioni sostenute dai passaggi recuperati, misurata su un insieme di test interno

TASSO DI ERRORE

Prove su casi storici anonimizzati, con soglie di accettazione definite prima dei test

REGRESSIONE

Ogni modifica a prompt, modello o indice comporta la riesecuzione della suite; i prompt si versionano come il codice

Dopo il rilascio

La valutazione prosegue con il campionamento umano periodico degli output, il monitoraggio della latenza e del drift del corpus e un canale di segnalazione per gli analisti.

Indagine sul sistema AI

Cosa registrare

- Prompt completi e contesti recuperati dal RAG
- Chiamate ai tool, con parametri ed esiti
- Versioni di modello, prompt di sistema e indice
- Identità del richiedente e marca temporale, con hash e conservazione da evidenza

Come si indaga

- Ricostruzione della catena causale: input, contesto, decisione, azione
- Per una prompt injection: individuare il documento vettore nell'audit trail
- Per una decisione errata: versione in uso, contesto fornito e output generato
- I log del sistema AI si preservano come ogni altra evidenza (Modulo 7)

Progettazione e analisi forense dell'AI



Retrieval-Augmented Generation nel SOC

La Retrieval-Augmented Generation (RAG) risponde a limiti noti dei modelli linguistici, che non conoscono i dati dell'organizzazione, come playbook, incidenti passati e architettura di rete, e quando l'informazione manca tendono a produrre risposte plausibili ma infondate. I documenti interni vengono suddivisi in frammenti (**chunk**), trasformati in embedding e indicizzati; a ogni domanda il sistema recupera i passaggi più pertinenti, li inserisce nel prompt e chiede al modello di rispondere solo sulla base di quelli, citandoli. Rispetto al fine-tuning, l'aggiornamento richiede una nuova indicizzazione, ogni risposta può riportare le fonti e, se l'indice conserva i permessi dei documenti, il retrieval li applica e ciascun utente ottiene risposte solo dai documenti che può consultare.

Scenario didattico

Un analista chiede come sia stato contenuto l'ultimo caso di beaconing su un server di produzione. Il retrieval individua il rapporto post-incidente del caso e la sezione pertinente del playbook; la risposta riassume i passi seguiti e indica i documenti di origine, che l'analista apre per la verifica.

Cautele

Se l'indice contiene documenti obsoleti, le risposte riportano indicazioni superate, pur con citazioni corrette. La base documentale diventa inoltre superficie d'attacco, perché chi può scriverci può inserire istruzioni ostili che il retrieval porta nel prompt (prompt injection indiretta, Modulo 13).

Valutazione continua di un sistema AI difensivo

La valutazione continua si basa su un **golden set**: domande e casi con risposta corretta nota, costruiti dagli analisti e rieseguiti a ogni modifica di modello, prompt o indice. Le metriche dipendono dal compito; per un RAG si misurano la groundedness delle affermazioni, la copertura delle fonti e il tasso di rifiuto corretto quando la risposta non è nei documenti. Si aggiungono il **red teaming** interno, con strumenti come Garak e PyRIT per verificare la resistenza a prompt injection e jailbreak, e la raccolta in produzione del riscontro degli analisti. Ogni risposta errata segnalata da un analista e confermata in revisione diventa un nuovo caso del golden set, secondo lo stesso principio delle lezioni apprese.

Scenario didattico

Si supponga che l'aggiornamento del modello di embedding migliori la metrica media ma riduca il recupero dei documenti relativi alla rete OT. La regressione emerge dai casi OT del golden set e non dalla media aggregata; senza quei casi sarebbe emersa durante un incidente su un impianto.

Limiti

Un LLM usato come valutatore (LLM-as-judge) esamina molti output, ma eredita i bias del modello giudice: preferisce le risposte più lunghe ed è sensibile all'ordine di presentazione (Zheng et al., 2023), e va quindi calibrato su un campione di giudizi umani.

Analisi forense dei sistemi AI

L'analisi forense dei sistemi AI, ambito in formazione, studia anzitutto il **sistema AI coinvolto in un incidente**, come una decisione errata con danno, un agente manipolato tramite prompt injection o un modello sottratto o avvelenato. Ricostruire che cosa ha fatto e su quali basi è possibile solo se esistono log di prompt e risposte con hash e marca temporale, la versione di modello e configurazione, lo stato dell'indice RAG al momento del fatto e il registro delle azioni dell'agente. L'altro oggetto sono le **evidenze generate dall'AI**: metadati e filigrane dei contenuti sintetici, credenziali di provenienza C2PA (Modulo 10). Vale il principio della forensic readiness (Modulo 10, approfondimento 5): i dati non registrati al momento dei fatti non si recuperano, quindi l'osservabilità si progetta prima dell'incidente.

Scenario didattico

Un agente del SOC chiude in una notte quaranta segnalazioni di phishing come benigne. L'analisi dei prompt registrati trova in tutte la stessa istruzione iniettata (Modulo 13), e i log indicano versione del modello e orari. Il confronto tra versioni dell'indice RAG individua il documento alterato, caricato il giorno prima da un account compromesso.

Metodo

I requisiti di progetto sono la registrazione di prompt e output su storage append-only, il versionamento di modello, prompt di sistema e indice, la correlazione tra agente e azioni nei log di audit e una conservazione coerente con gli obblighi applicabili. Per i sistemi ad alto rischio l'art. 12 dell'AI Act richiede che il sistema consenta la registrazione automatica degli eventi, obbligo che per l'allegato III si applica dal 2 dicembre 2027 (Modulo 15).

Chunking, retrieval e reranking in un RAG

La qualità di un RAG (approfondimento 1) dipende anzitutto dal **chunking**, che stabilisce come si suddividono i documenti: i frammenti devono conservare il contesto senza diluire la pertinenza, e i playbook si suddividono meglio per sezione logica che per numero fisso di caratteri. Il **retrieval** puramente vettoriale individua male le corrispondenze esatte, come un codice di errore o un indirizzo IP; per questo si usa spesso la ricerca ibrida, che combina similarità vettoriale e parole chiave. Il reranking impiega infine un secondo modello per riordinare i candidati recuperati in base alla pertinenza rispetto alla domanda, prima di passarli al modello generativo.

Esempio

Alla domanda sulla procedura per l'errore 0x80070005 la ricerca vettoriale restituisce documenti generici sugli errori di permesso. Con la componente per parole chiave il codice esatto porta in cima la pagina corretta, senza modifiche al corpus né al modello.

Metodo

Quando un RAG produce una risposta errata si verifica prima il retrieval: se i frammenti corretti non erano tra quelli recuperati, il problema riguarda indice o ricerca; se c'erano e la risposta li ignora, riguarda prompt o modello. La recall del retrieval va misurata nel golden set (approfondimento 2).

Osservabilità e registrazione nei sistemi AI

L'analisi forense dei sistemi AI (approfondimento 3) presuppone un'osservabilità progettata in anticipo. Per ogni interazione si registrano il prompt completo e la risposta, con hash e marca temporale, insieme alla versione del modello e del prompt di sistema in uso e ai documenti recuperati dal RAG. Si annotano anche le azioni dell'agente con il loro esito e il riscontro umano, cioè l'accettazione o la correzione della proposta. La conservazione avviene su storage **append-only**, perché un log modificabile dopo l'incidente è esposto a contestazioni sull'integrità. Poiché i prompt contengono dati d'indagine e dati personali (Modulo 15), il log è a sua volta un trattamento da proteggere, con accessi tracciati e tempi di conservazione motivati, e il principio di minimizzazione impone di limitare la registrazione a ciò che serve alla ricostruzione.

Scenario didattico

A distanza di sei mesi una decisione presa con il supporto del copilota viene contestata. Con una registrazione adeguata si ricostruiscono il prompt, la versione del modello, i documenti forniti dal RAG, la risposta generata e la correzione umana successiva.

Nota normativa

Per i sistemi ad alto rischio l'AI Act prevede la registrazione automatica degli eventi (art. 12) e la conservazione dei log per almeno sei mesi, salvo diversa disposizione, da parte di fornitori (art. 19) e deployer (art. 26); per i sistemi dell'allegato III questi obblighi si applicano dal 2 dicembre 2027.

Sviluppo interno o acquisto delle capacità AI

La maggior parte delle organizzazioni non addestra modelli propri e integra componenti esistenti, scegliendo per ogni capacità tra sviluppo interno e acquisto. Acquistare la funzione AI di un SIEM o di un EDR, o un servizio esterno, accelera l'adozione e lascia la manutenzione al fornitore, ma limita spiegabilità e controllo, crea dipendenza dal fornitore e può comportare il trasferimento di dati all'esterno. Un RAG su modello eseguito in locale, come quello del laboratorio, conserva il controllo dei dati e del comportamento al prezzo di costi di sviluppo e manutenzione. In genere si acquistano capacità mature e generiche come il filtraggio dello spam o la detection di base, e si sviluppano quelle che richiedono conoscenza aziendale e controllo del dato. In entrambi i casi restano necessari valutazione continua, osservabilità e un responsabile umano delle decisioni (Modulo 15).

Scenario didattico

Un SOC di medie dimensioni adotta la detection basata su machine learning del proprio EDR, capacità generica mantenuta dal fornitore, e sviluppa con un RAG locale l'assistente sui propri runbook e rapporti post-incidente, che contengono dati da non trasferire all'esterno.

Metodo

Prima dell'acquisto si chiede al fornitore dove e per quanto tempo sono trattati i dati, quale spiegabilità offre l'output, come gestisce falsi positivi e aggiornamenti del modello, se è possibile una valutazione sul traffico dell'organizzazione e quale piano è previsto in caso di modifica o cessazione del servizio.

Prototipo RAG e incidente simulato

Parte A: costruzione

1. Prototipo RAG su un corpus di playbook didattici, con modello locale
2. Logging completo: prompt, contesti, versioni
3. Mini-suite di valutazione: 10 domande con risposta attesa

Parte B: indagine

1. Il docente inietta un documento malevolo nel corpus (prompt injection)
2. Dall'audit trail: ricostruire vettore, dati esposti e azioni indebite
3. Mini-relazione forense dell'incidente AI

Lecture e autovalutazione

Lecture obbligatorie

- Lewis et al., *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*, NeurIPS 2020
- NIST, *AI RMF Playbook*
- NIST AI 600-1, *Generative Artificial Intelligence Profile*, 2024

Domande di autovalutazione

1. Quando preferire RAG al fine-tuning in un SOC?
2. Quali log rendono indagabile un agente LLM?
3. Come si definisce un test di regressione per i prompt?
4. Quali rischi introduce l'infrastruttura di inferenza stessa?
5. Come si preserva l'integrità probatoria dei log di un sistema AI?

FINE DELLA LEZIONE • MODULO 14

Grazie

Costruire l'AI difensiva e AI forensics • Corso «Incident Response potenziata dall'AI»

Domande e materiali: ricevimento del docente. Profilo e contatti:
vincenzocalabro.it



Vincenzo Calabrò

CYBER SECURITY • DIGITAL FORENSICS



VINCENZOCALABRO.IT