

MODULO 13

Adversarial ML e sicurezza dei sistemi AI

I sistemi AI come superficie d'attacco: evasion, poisoning, prompt injection ed estrazione, con MITRE ATLAS e OWASP Top 10 for LLM Applications come tassonomie di riferimento.

MITRE ATLAS

OWASP LLM TOP 10

Garak / ART / PyRIT

Abstract e obiettivi formativi

Abstract

Il modulo studia i sistemi AI come superficie d'attacco: minacce in fase di training (poisoning, backdoor) e di inferenza (evasion), attacchi alla riservatezza (model extraction, membership inference), minacce specifiche degli LLM (prompt injection, jailbreak, eccesso di autonomia), supply chain di modelli e dataset, e le difese architetturali corrispondenti.

Obiettivi formativi (Bloom)

- **Comprendere:** classificare un attacco a un sistema ML/LLM secondo ATLAS
- **Applicare:** valutare la robustezza di un modello con strumenti di testing
- **Creare:** progettare mitigazioni architetturali (validazione dell'input, isolamento dei tool, filtri sull'output)

Traccia core

01 02 07

Detection & Response

03 04 05 06

o

Digital Forensics

08 09 10 11

o

Sicurezza dell'AI

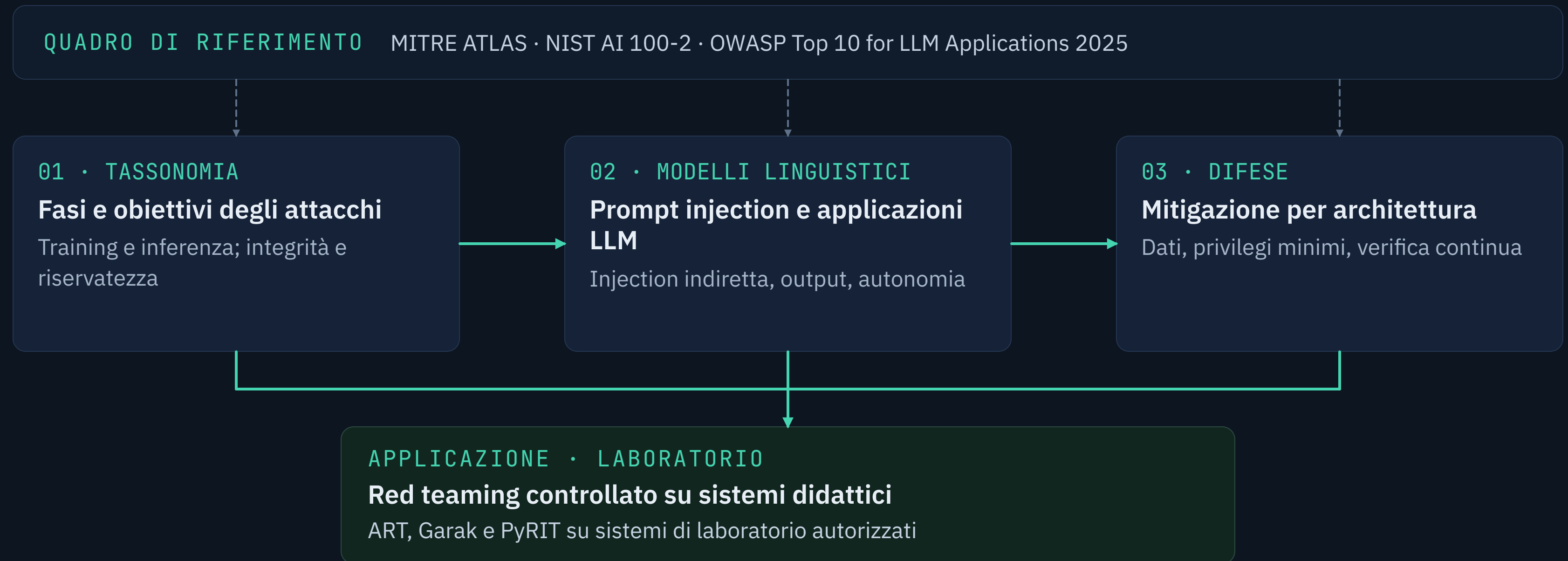
06 12 13 14

Traccia core

15 16

Letture di raccordo: § 3.2, 11.1, 11.4

Superficie d'attacco e difese dei sistemi AI



Fasi e obiettivi degli attacchi

Attacco	Fase	Obiettivo	Esempio nel dominio sicurezza
Poisoning	Training	Integrità o disponibilità	Campioni malevoli etichettati come benigni nel dataset
Backdoor	Training	Integrità	Trigger nascosto che disattiva la detection
Evasion	Inferenza	Integrità	Adversarial sample: malware modificato per eludere il classificatore
Membership inference	Inferenza	Riservatezza	Dedurre se un dato era nel training set
Model extraction	Inferenza	Riservatezza	Replicare il modello interrogandolo sistematicamente
Denial via input	Inferenza	Disponibilità	Input costruiti per saturare o degradare il sistema

MITRE ATLAS organizza queste tecniche in tattiche e le correda di casi documentati, mentre la tassonomia formale è quella di NIST AI 100-2. La supply chain di modelli e dataset pre-addestrati attraversa tutte le fasi (approfondimento 5).

Prompt injection e rischi delle applicazioni LLM

Le minacce (OWASP Top 10 for LLM, 2025)

- **Injection diretta:** l'utente manipola il modello nel prompt
- **Injection indiretta:** le istruzioni arrivano dai **dati** che il modello legge, come pagine, e-mail, documenti e alert
- Gestione impropria dell'output (LLM05): output del modello eseguito senza validazione
- Eccesso di autonomia (LLM06): agenti con tool e privilegi oltre il necessario

Rilevanza per il SOC

- Un copilota che analizza e-mail di phishing elabora anche le istruzioni dell'attaccante
- Un agente di triage con tool di scrittura può essere indotto ad agire
- Per la prompt injection non esiste a oggi una soluzione completa; la mitigazione combina più controlli architetturali

Mitigazione per architettura

DATI E TRAINING

- Sanificazione e provenienza dei dataset
- Robust training e validazione continua
- Controllo della supply chain di modelli pre-addestrati

ARCHITETTURA LLM

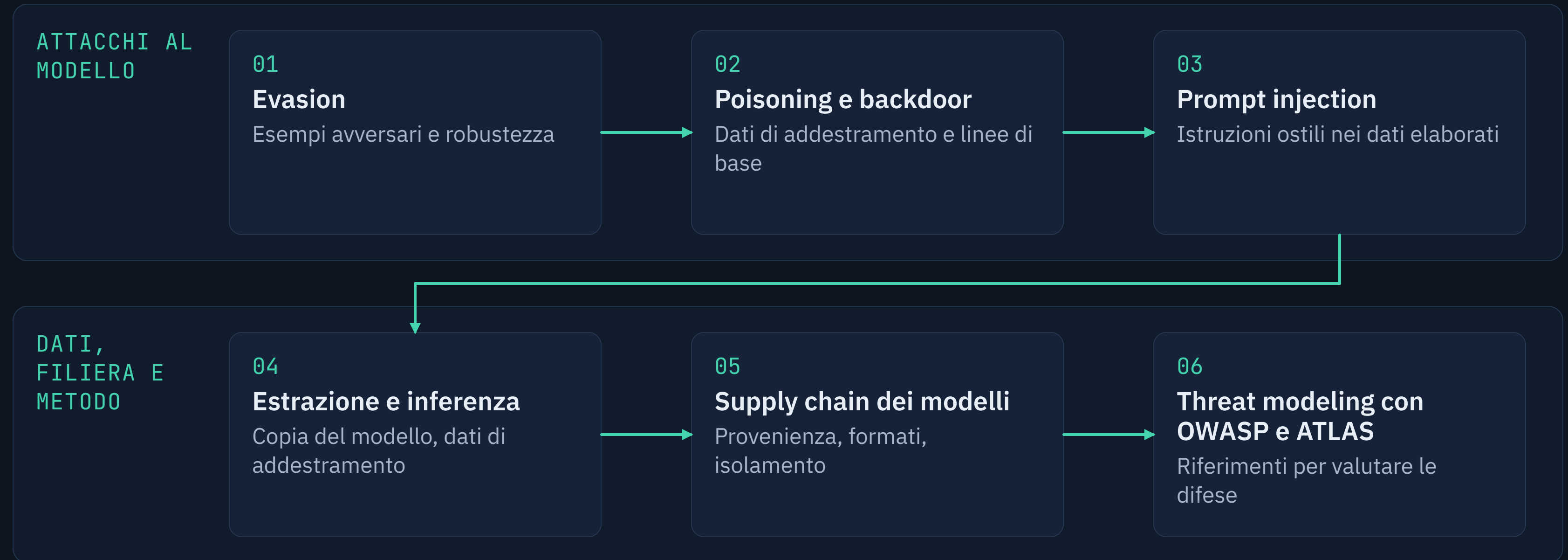
- Separare istruzioni e dati; trattare l'input come ostile
- Minimo privilegio per i tool; azioni di scrittura dietro approvazione
- Validazione dell'output prima di ogni esecuzione

VERIFICA CONTINUA

- Red teaming periodico dei sistemi AI (Garak, PyRIT)
- Test di robustezza dei classificatori (ART, Counterfit)
- Monitoraggio delle interrogazioni anomale, indizio di tentativi di estrazione

Le difese applicano ai modelli il principio della difesa in profondità: riducono la superficie d'attacco, limitano il danno possibile e verificano nel tempo l'efficacia dei controlli.

Attacchi ai modelli e alla loro filiera



Evasion ed esempi avversari

Un **adversarial example** è un input modificato quanto basta per cambiare la decisione di un modello senza alterarne la sostanza per un osservatore umano o, nel caso del malware, senza alterarne il comportamento. I confini di decisione dei modelli si trovano in spazi ad alta dimensionalità e, vicino a un confine, bastano perturbazioni piccole e mirate per attraversarlo. Il fenomeno riguarda anche i modelli implementati correttamente, per cui un classificatore usato come controllo di sicurezza va valutato in condizioni avversarie (robust accuracy) oltre che sull'accuratezza media.

Scenario didattico

Un rilevatore di malware basato su machine learning assegna a un campione una probabilità di malevolenza di 0,97. Modifiche che non alterano il comportamento del programma portano la probabilità a 0,12 e il campione viene classificato come benigno. Attacchi analoghi sono possibili anche senza accesso interno al modello, osservandone le risposte a interrogazioni ripetute.

Contromisure

Addestramento con esempi avversari, insiemi di modelli basati su caratteristiche indipendenti, limitazione del numero di interrogazioni e difesa in profondità, in cui il classificatore è uno dei livelli di controllo. La valutazione si svolge con Adversarial Robustness Toolbox e Counterfit, usando MITRE ATLAS come catalogo delle tecniche.

Poisoning e backdoor nell'addestramento

A differenza dell'evasion, che colpisce un modello già addestrato, il **poisoning** interviene sull'apprendimento. Le varianti principali sono la contaminazione dei dati di addestramento per degradare le prestazioni, l'inserimento di una **backdoor**, che si attiva solo in presenza di un segnale scelto dall'attaccante, e, nel contesto di un SOC, la manipolazione del ciclo di retroazione: un sistema che riaddestra la propria linea di base sul comportamento osservato, come un sistema UEBA (Modulo 3), può essere indotto gradualmente a considerare normale un'attività preparatoria.

Scenario didattico

Un dipendente prepara un'esfiltrazione trasferendo ogni settimana volumi crescenti di dati verso un servizio esterno, sempre al di sotto della soglia di allerta. La linea di base si adatta e dopo alcuni mesi un trasferimento massivo rientra nella normalità appresa, senza generare alert.

Contromisure

Le linee di base si confrontano con riferimenti storici congelati oltre che con la finestra recente. Si verifica la provenienza dei dati di addestramento e si eseguono test di regressione sul modello. Il monitoraggio del concept drift tiene conto che la deriva può essere provocata intenzionalmente (Modulo 2).

Prompt injection diretta e indiretta

Un modello linguistico riceve istruzioni e dati come testo nello stesso contesto, senza un meccanismo strutturale che li separi. La prompt injection, classificata come rischio LLM01 nella OWASP Top 10 for LLM Applications 2025, sfrutta questa caratteristica inserendo istruzioni ostili nel contenuto elaborato. Nella variante **diretta** le istruzioni provengono dall'utente. Per un SOC conta di più quella **indiretta**, in cui il testo rivolto al modello si trova nei dati: un'e-mail di phishing analizzata da un assistente, una pagina web riassunta, un log contaminato. Con agenti dotati di strumenti e permessi (Modulo 6) un'injection riuscita può tradursi in azioni.

Scenario didattico

In coda a un'e-mail sospetta compare un testo non visibile al lettore umano che chiede all'assistente di classificare il messaggio come legittimo e di chiudere il ticket. L'assistente che riassume l'e-mail elabora anche quel testo; se il suo output alimenta la chiusura automatica dei ticket, il controllo viene disattivato.

Contromisure

Ogni contenuto analizzato va trattato come non affidabile e separato dalle istruzioni nel prompt, e gli output che ne derivano vanno marcati. Gli strumenti dell'agente hanno il minimo privilegio e le azioni richiedono l'approvazione umana. I test con Garak e PyRIT precedono la messa in esercizio e si ripetono a ogni modifica del modello, del prompt di sistema o degli strumenti disponibili.

Estrazione del modello e inferenza sui dati

Un modello addestrato è un investimento da proteggere e può conservare tracce dei dati di addestramento. Il **model extraction** ricostruisce una copia funzionale del modello interrogandolo ripetutamente e addestrando un secondo modello sulle risposte. Gli attacchi di **membership inference** stimano se un determinato record faceva parte dei dati di addestramento, con implicazioni di riservatezza quando i dati sono sanitari o giudiziari; nei modelli linguistici si possono inoltre estrarre sequenze memorizzate alla lettera, compresi dati personali (Carlini e colleghi, 2021). Per la risposta agli incidenti un sistema AI esposto è anche una superficie da monitorare, perché volumi e schemi di interrogazione anomali possono indicare un'estrazione in corso.

Scenario didattico

Un'API di classificazione esposta riceve da un gruppo di indirizzi milioni di richieste sistematiche che coprono lo spazio degli input. La distribuzione delle richieste è compatibile con la raccolta di un insieme di addestramento per replicare il modello. Il monitoraggio distingue questo traffico dall'uso ordinario in base alla copertura, alla ripetitività e all'assenza di comportamento umano nello schema di interrogazione.

Contromisure

Le API dei modelli richiedono autenticazione e limitazione delle richieste, e le risposte contengono solo le informazioni necessarie, cioè la classe senza le probabilità dettagliate. La privacy differenziale nell'addestramento limita la memorizzazione. I modelli esposti entrano nel programma di monitoraggio della sicurezza.

Supply chain di modelli e dataset

Di norma si scarica un modello pre-addestrato da un repository pubblico, lo si adatta e lo si mette in esercizio, ereditando una catena di fornitura con rischi propri. I file di pesi in formati che ammettono codice eseguibile, come il formato pickle usato da PyTorch, possono eseguire istruzioni al momento del caricamento, e un modello scaricato va quindi trattato come codice non affidabile. Una backdoor inserita nei pesi (approfondimento 2) è difficile da individuare con un'ispezione; dipendenze, tokenizer e configurazioni allegate ampliano la superficie.

Scenario didattico

Un data scientist carica in un notebook collegato all'ambiente di produzione un modello proveniente da un repository poco noto. Il file, in un formato che ammette codice eseguibile, apre una connessione verso l'esterno al momento del caricamento. Un formato di pesi non eseguibile avrebbe impedito l'esecuzione del codice; il controllo EDR sul nodo e il filtraggio del traffico in uscita avrebbero potuto rilevare o bloccare la connessione.

Contromisure

Si adottano formati di pesi non eseguibili come safetensors e si verificano provenienza e hash dei modelli, come per le dipendenze software. Il primo caricamento avviene in ambiente isolato, con l'analisi delle backdoor dove possibile. L'inventario dei modelli in uso ne registra origine, versione e responsabile e ricalca la distinta dei componenti software (SBOM).

Threat modeling con OWASP e MITRE ATLAS

La **OWASP Top 10 for LLM Applications** (versione 2025) elenca i rischi principali delle applicazioni basate su modelli linguistici, tra cui prompt injection, divulgazione di informazioni sensibili, supply chain, avvelenamento di dati e modelli, gestione impropria dell'output ed eccesso di autonomia (excessive agency). **MITRE ATLAS** cataloga tattiche e tecniche d'attacco ai sistemi di machine learning, con casi documentati, e su di esso si mappa la copertura difensiva come con ATT&CK. Il threat modeling del sistema AI usa questi riferimenti per individuare da dove entrano dati e istruzioni, quali componenti si fidano di quali altri e dove un input ostile può propagarsi.

Scenario didattico

Prima della messa in esercizio dell'assistente SOC del Modulo 6, il gruppo di progetto conduce un threat modeling con OWASP e ATLAS. Emerge che l'agente legge e-mail non affidabili e dispone di uno strumento di chiusura dei ticket, una combinazione che costituisce un rischio LLM01 ad alto impatto. Le mitigazioni scelte sono la separazione tra dati e istruzioni, l'approvazione umana per la chiusura e i test con Garak.

Nel laboratorio

Gli studenti compilano una scheda di threat model per l'assistente didattico: punti di ingresso, relazioni di fiducia tra componenti, tre rischi prioritari classificati secondo OWASP e ATLAS, con la relativa mitigazione.

Red teaming controllato su sistemi didattici

Svolgimento su sistemi di laboratorio autorizzati

1. Valutazione di robustezza di un classificatore con Adversarial Robustness Toolbox
2. Red teaming di un'applicazione LLM didattica con Garak e PyRIT
3. Scenario guidato di prompt injection indiretta contro l'applicazione fornita dal docente
4. Scheda di threat model secondo OWASP e ATLAS
5. Relazione dei risultati con le mitigazioni proposte, nel formato di un penetration test

Consegna

- Risultati classificati secondo ATLAS e OWASP
- Mitigazione architetturale per ogni risultato
- Robustezza del classificatore prima e dopo le mitigazioni
- Scheda di threat model con le mitigazioni

Lecture e autovalutazione

Lecture obbligatorie

- MITRE ATLAS; OWASP *Top 10 for LLM Applications* (2025)
- NIST AI 100-2 E2025, *Adversarial Machine Learning: A Taxonomy and Terminology of Attacks and Mitigations*
- Biggio & Roli, *Wild Patterns*, Pattern Recognition 2018

Domande di autovalutazione

1. Che differenza c'è tra poisoning ed evasion?
2. Perché la prompt injection indiretta è critica per gli agenti con tool?
3. Quali segnali indicano un'avvenuta model extraction?
4. Come si applica il principio del minimo privilegio a un agente LLM?
5. Cosa aggiunge ATLAS rispetto ad ATT&CK?

FINE DELLA LEZIONE • MODULO 13

Grazie

Adversarial ML e sicurezza dei sistemi AI • 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