

Incident Response potenziata dall'AI

Dispensa del corso · 16 capitoli

Vincenzo Calabrò · vincenzocalabro.it

Edizione luglio 2026

Indice

Introduzione	3
Capitolo 1. Fondamenti di Incident Response	5
Capitolo 2. Introduzione all'AI per la sicurezza	13
Capitolo 3. Detection potenziata dall'AI	21
Capitolo 4. Threat Intelligence e AI	28
Capitolo 5. SOAR e automazione della risposta	35
Capitolo 6. LLM e agenti nel SOC	43
Capitolo 7. Fondamenti di Digital Forensics	50
Capitolo 8. Digital Forensics assistita dall'AI	58
Capitolo 9. Memory e network forensics con AI	66
Capitolo 10. Mobile, cloud e multimedia forensics	74
Capitolo 11. Malware analysis con AI	82
Capitolo 12. AI offensiva e minacce emergenti	90
Capitolo 13. Adversarial ML e sicurezza dei sistemi AI	98
Capitolo 14. Costruire l'AI difensiva e AI forensics	107
Capitolo 15. Governance, etica e diritto	115
Capitolo 16. Project work finale	125
Glossario	134
Bibliografia e fonti	144

Introduzione

Finalità della dispensa

La dispensa accompagna il corso «Incident Response potenziata dall'AI» e ne raccoglie i contenuti in forma estesa. Le slide delle lezioni presentano i concetti in modo sintetico e servono a guidare la discussione in aula; la dispensa sviluppa gli stessi argomenti con definizioni complete, riferimenti normativi e tecnici, esempi commentati e indicazioni per lo studio individuale.

Il corso tratta la risposta agli incidenti informatici e l'analisi forense digitale nel contesto attuale, in cui i sistemi di intelligenza artificiale intervengono in tre modi distinti: come strumenti a supporto dell'analista (rilevamento, triage, analisi di grandi volumi di dati), come mezzi a disposizione degli attaccanti e come sistemi da proteggere e, se necessario, da sottoporre a indagine. Ogni capitolo tiene separati questi tre piani e indica, per ciascuno, contributi e limiti.

Struttura

La dispensa contiene sedici capitoli, uno per modulo del corso. I moduli formano una traccia core, comune a tutti gli studenti, e tre specializzazioni, tra le quali ogni studente sceglie:

- traccia core: capitoli 1, 2 e 7 in apertura, capitoli 15 e 16 in chiusura, con il processo di risposta, le basi dell'AI per la sicurezza, il metodo forense, il quadro giuridico e il project work finale;
- specializzazione Detection & Response, capitoli 3–6: detection, threat intelligence, automazione della risposta, modelli linguistici e agenti nel SOC;
- specializzazione Digital Forensics, capitoli 8–11: forensics assistita dall'AI, memoria e rete, dispositivi mobili, cloud e contenuti multimediali, malware;
- specializzazione Sicurezza dell'AI, capitoli 6 e 12–14: modelli linguistici e agenti nel SOC, poi l'AI come strumento offensivo, come superficie d'attacco e come oggetto di indagine, con la progettazione di sistemi difensivi.

Lo studente segue i capitoli 1, 2 e 7, poi quelli della propria specializzazione, infine i capitoli 15 e 16. Il capitolo 6 appartiene a due specializzazioni, Detection & Response e Sicurezza dell'AI. Quando un capitolo presuppone sezioni di un'altra traccia, la riga «Percorso» che precede gli obiettivi le indica come letture di raccordo, da studiare prima del capitolo. Gli altri rimandi a capitoli di altre tracce sono indicazioni di approfondimento facoltativo. Le letture di raccordo non hanno una prova propria: le loro nozioni rientrano nelle domande della prova scritta sui capitoli che le usano.

Ogni capitolo segue lo stesso schema: percorso, obiettivi di apprendimento, tre sezioni teoriche, sei approfondimenti, la descrizione del laboratorio, le letture e cinque domande di autovalutazione con traccia di risposta. Il capitolo 16, dedicato al project work, ha quattro sezioni descrittive.

tive e nessun laboratorio separato, perché la simulazione finale è essa stessa l'attività di laboratorio. La numerazione delle sezioni corrisponde all'ordine delle slide; i riferimenti interni usano la forma «§ 7.6» o «capitolo 7».

Prerequisiti

Si presuppone la conoscenza di base di reti TCP/IP, sistemi operativi Windows e Linux e principi di sicurezza informatica (riservatezza, integrità, disponibilità; autenticazione e controllo degli accessi). Per i capitoli sul machine learning è utile una familiarità elementare con la statistica descrittiva e con la probabilità condizionata.

Convenzioni

Gli esempi sono di tre tipi:

- **esempi** in senso stretto: illustrano una definizione in poche righe, senza riferimento a un caso concreto;
- **scenari didattici**: casi costruiti per illustrare un concetto; nomi, orari e numeri fanno parte dell'ipotesi e non descrivono fatti reali;
- **casi documentati**: eventi effettivamente accaduti, accompagnati dall'anno e da una fonte pubblica.

Le statistiche generali compaiono solo con l'indicazione della fonte.

I riferimenti normativi sono aggiornati a luglio 2026. Norme, linee guida e standard vengono modificati con frequenza; prima di applicarli a un caso concreto occorre verificarne la versione vigente sulle fonti ufficiali (Gazzetta Ufficiale dell'Unione europea, Gazzetta Ufficiale della Repubblica italiana, siti di NIST, ISO, ENISA e ACN).

Il glossario finale raccoglie i termini tecnici principali; la bibliografia riunisce le letture indicate nei singoli capitoli.

Capitolo 1. Fondamenti di Incident Response

Il capitolo introduce la risposta agli incidenti come processo strutturato e ripetibile. Ne descrive il ciclo di vita secondo i principali modelli di riferimento, le strutture e i documenti che lo rendono operativo, le metriche con cui se ne misura l'efficacia e gli obblighi di notifica con i relativi termini. Gli approfondimenti precisano alcune nozioni che il resto del corso dà per acquisite: la distinzione tra evento, alert e incidente, le strategie di contenimento, il calcolo delle metriche, la classificazione per severità e la comunicazione durante una crisi. Il ruolo dell'intelligenza artificiale è appena accennato: dal capitolo 2 in poi ogni strumento basato sull'AI verrà collocato in una delle fasi qui descritte.

Percorso. Traccia core, primo capitolo del corso.

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- descrivere le fasi del ciclo di vita di un incidente e le differenze tra i modelli SANS, NIST e ISO/IEC;
- classificare un evento per severità e categoria, motivando la scelta;
- progettare un playbook per uno scenario assegnato, distinguendolo dal runbook che lo rende eseguibile.

1.1 Il ciclo di vita dell'incidente

La gestione di un incidente si descrive come una sequenza di fasi. Un modello molto diffuso nella formazione professionale è quello del SANS Institute, noto con l'acronimo PICERL: preparazione (Preparation), identificazione (Identification), contenimento (Containment), eradicazione (Eradication), ripristino (Recovery) e lezioni apprese (Lessons learned).

Le fasi hanno il seguente contenuto:

- **preparazione:** policy, ruoli assegnati, strumenti di raccolta e analisi, canali di comunicazione, playbook, formazione ed esercitazioni, cioè tutto ciò che deve esistere prima dell'incidente;
- **identificazione:** rilevazione dei segnali anomali, loro valutazione (triage) e decisione sulla dichiarazione di incidente;
- **contenimento:** limitazione della propagazione e del danno;
- **eradicazione:** rimozione della causa (malware, account compromessi, vulnerabilità sfruttate);
- **ripristino:** ritorno in esercizio dei sistemi sotto monitoraggio rafforzato;
- **lezioni apprese:** revisione post-incidente che individua le cause e aggiorna la preparazione.

Il NIST ha descritto lo stesso processo nella SP 800-61 Rev. 2 (2012) raggruppando le fasi in quattro: preparazione; rilevamento e analisi; contenimento, eradicazione e ripristino; attività post-incidente. La Rev. 3, pubblicata nell'aprile 2025, sostituisce il modello a quattro fasi con un

ciclo di vita basato sulle sei funzioni del Cybersecurity Framework 2.0: Govern, Identify e Protect costituiscono la preparazione, Detect, Respond e Recover la risposta all'incidente; le lezioni apprese confluiscono nella categoria Improvement (ID.IM) della funzione Identify e alimentano tutte le funzioni. La norma ISO/IEC 27035-1:2023 propone a sua volta cinque fasi: pianificazione e preparazione, rilevamento e segnalazione, valutazione e decisione, risposta, lezioni apprese.

Le differenze tra i modelli riguardano soprattutto il raggruppamento delle attività. In tutti i modelli il contenimento precede l'eradicazione, perché rimuovere la causa senza aver bloccato la propagazione espone a reinfezioni, e le azioni di risposta vanno scelte in modo da preservare le evidenze. Le lezioni apprese sono una fase del processo, con attività e responsabili definiti. Il rapporto tra contenimento e conservazione delle evidenze viene ripreso nei capitoli dedicati alla digital forensics (capitoli 7–10).

1.2 Strutture, ruoli, playbook e runbook

La risposta agli incidenti richiede strutture con compiti distinti. Il **SOC** (Security Operations Center) svolge il monitoraggio continuo: raccoglie e correla eventi, gestisce gli alert e ne effettua il primo triage, spesso con analisti organizzati su più livelli. Il **CSIRT** (Computer Security Incident Response Team) è il gruppo che gestisce l'incidente dichiarato: coordina le attività tecniche, le decisioni e le comunicazioni. Il termine CERT (Computer Emergency Response Team) risale al CERT Coordination Center, istituito nel 1988 presso la Carnegie Mellon University dopo il worm Morris; è un marchio registrato della stessa università ed è usato come denominazione da molti CSIRT nazionali, settoriali o aziendali, mentre CSIRT è il termine generico. In Italia il CSIRT nazionale (CSIRT Italia) opera presso l'Agenzia per la cybersicurezza nazionale (ACN). Molte organizzazioni affidano parte di queste funzioni a fornitori esterni (servizi gestiti di sicurezza o di detection and response): in tal caso contratti e procedure devono stabilire a chi spettano le decisioni, le comunicazioni e la conservazione delle evidenze.

Nella gestione di un incidente il coordinamento e le decisioni spettano al responsabile della gestione (incident commander o incident manager), affiancato da analisti tecnici, specialisti forensi e referenti per la comunicazione, per la funzione legale e per la protezione dei dati (DPO). Le catene di escalation e le regole di comunicazione di crisi vanno definite in anticipo.

Il **playbook** descrive la strategia di risposta per una categoria di incidenti: fasi, decisioni da prendere, responsabili, criteri di escalation, momento in cui coinvolgere la funzione legale e la comunicazione. Il **runbook** descrive invece la procedura operativa che un analista esegue passo per passo: comandi, strumenti, verifiche, moduli da compilare. Il runbook è la parte del processo che si presta più direttamente all'automazione; nelle piattaforme SOAR i flussi automatizzati prendono anch'essi il nome di playbook e traducono in sequenze eseguibili le decisioni del playbook e i passi del runbook (capitolo 5). Entrambi i documenti si gestiscono sotto controllo di versione, si verificano nelle esercitazioni e si aggiornano dopo ogni incidente.

1.3 Metriche di risposta e obblighi di notifica

Le metriche più usate per misurare l'efficacia del processo sono:

- **MTTD** (Mean Time To Detect): tempo medio tra l'inizio di un incidente e il suo rilevamento;
- **MTTR**: tempo medio di risposta, di ripristino o di rimedio, a seconda delle organizzazioni; va quindi sempre accompagnato dalla definizione adottata;
- **dwell time**: permanenza dell'attaccante nell'ambiente prima della scoperta.

Il rapporto annuale M-Trends di Mandiant pubblica il valore mediano globale del dwell time: 10 giorni per le indagini del 2023, 11 per quelle del 2024 e 14 per quelle del 2025 (edizione 2026). Il valore dipende molto dalla modalità di scoperta: interna, segnalazione da parte di soggetti esterni, notifica da parte dello stesso attaccante, tipica dei casi di ransomware.

Il processo è vincolato da termini fissati dalle norme. Il Regolamento (UE) 2016/679 (GDPR) prevede all'art. 33 che il titolare notifichi una violazione di dati personali all'autorità di controllo senza ingiustificato ritardo e, ove possibile, entro 72 ore dal momento in cui ne è venuto a conoscenza, salvo che sia improbabile che la violazione presenti un rischio per i diritti e le libertà delle persone. L'art. 34 impone la comunicazione agli interessati quando il rischio è elevato. Ogni violazione va comunque documentata (art. 33, par. 5). Nel novembre 2025 la Commissione europea ha proposto, nel pacchetto di semplificazione noto come «Digital Omnibus», di modificare l'art. 33 limitando la notifica all'autorità alle violazioni a rischio elevato, portando il termine a 96 ore e introducendo un punto di accesso unico per le notifiche. A luglio 2026 questa parte del pacchetto è ancora in negoziato tra Parlamento e Consiglio e si applica il testo vigente dell'art. 33; la parte relativa all'AI Act è stata invece adottata con il Regolamento (UE) 2026/1744 (capitolo 15). Prima di applicare la disciplina a un caso concreto occorre verificare lo stato dell'iter legislativo.

La Direttiva (UE) 2022/2555 (NIS2), recepita in Italia con il D.Lgs. 4 settembre 2024, n. 138, si applica ai soggetti essenziali e importanti e prevede per gli incidenti significativi una notifica in più fasi al CSIRT Italia (art. 25, comma 5): una pre-notifica entro 24 ore da quando si è venuti a conoscenza dell'incidente, una notifica entro 72 ore dallo stesso momento, una relazione intermedia se richiesta e una relazione finale entro un mese dalla trasmissione della notifica; se l'incidente è ancora in corso, la relazione finale segue entro un mese dalla conclusione della sua gestione. Esistono inoltre regimi settoriali, come quello previsto per il settore finanziario dal Regolamento (UE) 2022/2554 (DORA). Un incidente può far scattare più obblighi contemporaneamente, con destinatari e termini diversi; il playbook deve indicare chi valuta ciascun obbligo e su quali informazioni.

1.4 Evento, alert e incidente

Un **evento** è qualunque occorrenza osservabile in un sistema o in una rete: un accesso, una connessione, l'avvio di un processo. Un **alert** è un evento, o una combinazione di eventi, che una regola o un modello segnala come potenzialmente rilevante. Un incidente, nella definizione adottata dal NIST (SP 800-61r3, che riprende il FISMA 2014), è un'occorrenza che, senza autorizzazione legittima, mette in pericolo in modo effettivo o imminente l'integrità, la riservatezza o la disponibilità di informazioni o di un sistema informativo, oppure costituisce una violazione, o una minaccia imminente di violazione, di leggi, policy o procedure di sicurezza o regole di uso accettabile.

Il passaggio da alert a incidente richiede una decisione esplicita e documentata, presa secondo criteri predefiniti. La dichiarazione di incidente è un atto formale: attiva il CSIRT, avvia la valutazione degli obblighi di notifica e impone la conservazione delle evidenze. I termini di notifica decorrono invece dal momento in cui l'organizzazione viene a conoscenza della violazione o dell'incidente significativo, che può precedere la dichiarazione formale.

Scenario didattico. Diecimila accessi falliti in un'ora su un portale esposto, distribuiti su molti account con poche password comuni, sono diecimila eventi e producono un alert di password spraying. Si ha un incidente quando uno degli accessi riesce: la compromissione di un account viola le policy di sicurezza e mette in pericolo la riservatezza delle informazioni; se l'account è amministrativo, la severità aumenta. Severità e categoria assegnate in quel momento determinano escalation e risorse e orientano la verifica degli obblighi di notifica, i cui presupposti sono fissati dalle norme.

1.5 Contenimento a breve e a lungo termine

Il **contenimento a breve termine** limita subito il danno con azioni rapide e possibilmente reversibili: isolamento degli host dalla rete, blocco o sospensione degli account, segmentazione d'emergenza, blocco di indirizzi e domini. Queste azioni vanno scelte in modo da preservare le evidenze: un host la cui memoria è necessaria all'indagine non va spento prima dell'acquisizione (capitolo 7). Il **contenimento a lungo termine** mantiene l'operatività mentre si prepara l'eradicazione: correttivi temporanei sui sistemi che restano in esercizio (rimozione degli account e delle backdoor dell'attaccante, patch sui sistemi colpiti e su quelli adiacenti), rotazione delle credenziali, monitoraggio rafforzato delle vie d'ingresso note; in parallelo si ricostruiscono sistemi puliti.

Scenario didattico. In un attacco ransomware, nel breve termine si isolano i segmenti colpiti e si sospendono le condivisioni di rete, lasciando accesi gli host necessari all'acquisizione della memoria. Nel lungo termine si applicano correttivi temporanei ai sistemi che devono restare in esercizio e si ricostruisce in parallelo un'infrastruttura pulita; il ritorno dei servizi da backup verificati avviene nella fase di ripristino, mentre l'ambiente compromesso resta disponibile per l'analisi.

La scelta del momento del contenimento comporta un compromesso. Un contenimento precoce limita il danno ma può avvisare l'attaccante, che potrebbe accelerare l'attacco o cancellare le tracce. Un'osservazione più lunga chiarisce l'estensione dell'intrusione al prezzo di un'esposizione prolungata. La decisione dipende dal caso e va motivata e registrata, perché sarà esaminata nella revisione post-incidente e, se necessario, in sede giudiziaria.

1.6 Calcolo delle metriche su un caso

Scenario didattico. Un attaccante accede alla rete lunedì alle 09:00 tramite un messaggio di phishing. Un alert del sistema di analisi comportamentale scatta mercoledì alle 15:00; il CSIRT dichiara l'incidente alle 16:30 dello stesso giorno e completa contenimento e ripristino venerdì alle 12:00.

Il dwell time, misurato dall'accesso alla scoperta, è di 54 ore. Il tempo dalla dichiarazione al ripristino è di 43,5 ore. MTDD e MTTR sono medie (o mediane, se la distribuzione è asimmetrica) calcolate sull'insieme degli incidenti di un periodo e non sul singolo caso. La definizione degli istanti di inizio e di fine fa parte della metrica: l'inizio dell'incidente, per esempio, è spesso noto solo a posteriori grazie all'analisi forense, e il valore del MTDD cambia se lo si misura dall'accesso iniziale o dalla prima azione rilevabile.

Le metriche servono a confrontare il processo con sé stesso nel tempo e a valutare gli investimenti, per esempio a verificare se l'introduzione del triage assistito dall'AI riduce il MTDD (capitolo 16). Quando diventano obiettivi individuali inducono i comportamenti opportunistici descritti dalla legge di Goodhart, come la chiusura prematura dei ticket o il declassamento di incidenti a eventi.

1.7 Playbook e runbook nella preparazione

Playbook e runbook, distinti nel § 1.2, si aggiornano con ritmi diversi: il playbook si rivede quando cambiano l'organizzazione, i fornitori o le norme, mentre il runbook segue l'evoluzione degli strumenti e dell'infrastruttura. Entrambi vanno redatti prima dell'incidente, verificati con esercitazioni e mantenuti sotto controllo di versione; durante la crisi si eseguono e, se necessario, si adattano registrando le deviazioni.

Scenario didattico. Il playbook ransomware di un'organizzazione assegna all'incident commander la scelta tra spegnimento e isolamento degli host, secondo criteri scritti (cifatura in corso, presenza di evidenze volatili). Il runbook collegato elenca i comandi di isolamento per la piattaforma EDR, per gli switch e per gli ambienti cloud. Durante un'esercitazione emerge che manca la procedura per il segmento di rete industriale (OT), e la correzione riguarda il runbook mentre il playbook resta valido.

Playbook scritti con precisione sono il presupposto dell'automazione trattata nei capitoli 5 e 6, perché un playbook ambiguo, una volta automatizzato, produce decisioni incoerenti su larga scala. Un modello linguistico può proporre bozze e segnalare disallineamenti tra runbook e documentazione tecnica.

1.8 Severità e triage degli incidenti

La severità di un incidente combina l'**impatto**, che riguarda dati, servizi, persone e obblighi normativi coinvolti, e l'**urgenza**, legata alla dinamica dell'attacco (in corso o concluso, in propagazione o circoscritto). Una matrice di impatto e urgenza, definita nel playbook, stabilisce

priorità, tempi di presa in carico e catena di reperibilità.

La classificazione iniziale è provvisoria. Quando l'indagine chiarisce l'estensione, un phishing isolato può rivelarsi una compromissione del dominio, e la severità va rivista. Il processo deve indicare chi rivaluta la severità, in quali momenti e con quale registrazione della decisione.

Scenario didattico. Alle 02:10 scatta un alert su un account di servizio con accesso al gestionale: impatto potenziale alto, nessuna attività in corso. Il caso è classificato di severità 3 e messo in coda. Alle 02:40 lo stesso account crea un nuovo utente amministrativo: l'urgenza diventa massima, la severità sale a 1 e si attiva la reperibilità.

I modelli di scoring trattati nel capitolo 3 propongono una priorità, ma il riferimento resta la matrice scritta: se il modello assegna una severità inferiore a quella prevista dalla matrice, prevale la matrice e la discrepanza viene registrata come possibile errore del modello.

1.9 Comunicazione durante la crisi

Ogni incidente grave procede su due linee parallele, quella tecnica e quella della comunicazione. Gli interlocutori hanno esigenze diverse. Il vertice aziendale chiede l'impatto e le decisioni da prendere, i gruppi tecnici i dettagli operativi. La funzione legale controlla il linguaggio, perché termini come «violazione» o affermazioni sulle cause possono avere conseguenze giuridiche. Con le autorità valgono contenuti e termini fissati dalle norme (capitolo 15), mentre ai dipendenti servono istruzioni semplici e immediate.

Le prassi più diffuse prevedono un solo portavoce autorizzato per ciascun canale, aggiornamenti a cadenza fissa anche in assenza di novità, la separazione tra fatti accertati e ipotesi e l'attenzione a non scrivere, nemmeno nelle chat interne, ipotesi non verificate o ammissioni che sarebbero pregiudizievoli se acquisite in un procedimento giudiziario.

Scenario didattico. Un ransomware viene rilevato il venerdì sera. Alle 21:00 il vertice riceve una nota di mezza pagina con i fatti accertati, l'impatto stimato e l'orario del prossimo aggiornamento, mentre i dipendenti ricevono l'istruzione di non accendere i portatili; i dettagli tecnici restano interni finché l'estensione non è accertata. Il titolare, con il supporto del DPO, avvia la valutazione prevista dall'art. 33 GDPR.

Un modello linguistico può preparare rapidamente versioni dello stesso aggiornamento per destinatari diversi. Ogni testo diretto all'esterno o alle autorità richiede comunque la revisione umana e legale, perché un'informazione errata in una notifica resta agli atti dell'autorità e può esporre l'organizzazione a contestazioni.

1.10 Laboratorio: tabletop su uno scenario ransomware

L'esercitazione tabletop simula la gestione di un incidente attraverso la discussione, senza intervenire su sistemi reali. Il docente assegna i ruoli (responsabile della gestione, analisti, comunicazione, funzione legale) e introduce progressivamente gli eventi dello scenario (inject), per

esempio la scoperta di file cifrati, una richiesta di riscatto, la segnalazione di dati pubblicati, una telefonata della stampa.

Il gruppo applica il ciclo PICERL: per ogni inject decide le azioni, ne indica il motivo e le registra con l'orario. Al termine si redige il verbale dell'incidente e si svolge un debriefing in cui ciascun ruolo riporta una lezione appresa.

Consegna: verbale d'incidente del gruppo; decisioni di notifica motivate ai sensi del GDPR e della disciplina NIS2; una lezione appresa per ruolo. Lo stesso scenario, arricchito con log sintetici realistici, è la base del project work (capitolo 16).

1.11 Letture

- NIST, *SP 800-61 Rev. 3, Incident Response Recommendations and Considerations for Cybersecurity Risk Management: A CSF 2.0 Community Profile*, 2025.
- NIST, *SP 800-61 Rev. 2, Computer Security Incident Handling Guide*, 2012 (ritirata il 3 aprile 2025 e sostituita dalla Rev. 3; utile per il modello a quattro fasi).
- P. Kral, *Incident Handler's Handbook*, SANS Institute, 2012.
- ISO/IEC 27035-1:2023, *Information security incident management, Part 1: Principles and process*.
- ENISA, *Good Practice Guide for Incident Management*, 2010.
- Regolamento (UE) 2016/679 (GDPR), artt. 33–34; Direttiva (UE) 2022/2555 (NIS2); D.Lgs. 4 settembre 2024, n. 138.

1.12 Domande di autovalutazione

1. Quali criteri distinguono un evento da un incidente? *Traccia.* L'evento è un'occorrenza osservabile; l'incidente compromette o minaccia in modo imminente riservatezza, integrità o disponibilità, oppure viola norme o policy. Il passaggio è una decisione formale, presa con criteri scritti, che attiva il CSIRT, la valutazione degli obblighi di notifica e la conservazione delle evidenze; i termini di notifica decorrono dalla conoscenza del fatto.
2. Perché il contenimento precede l'eradicazione? *Traccia.* Rimuovere la causa mentre l'attaccante può ancora muoversi espone a reinfezioni e a nuovi danni; il contenimento blocca la propagazione e crea le condizioni per un'eradicazione completa, preservando le evidenze necessarie all'indagine.
3. Quali metriche misurano l'efficacia di un CSIRT? *Traccia.* MTTD, MTTR (con la definizione adottata), dwell time, tasso di riapertura dei casi, rispetto dei termini di notifica. Le metriche vanno calcolate sull'insieme degli incidenti e usate per migliorare il processo; se diventano obiettivi individuali inducono comportamenti opportunistici (legge di Goodhart).
4. Quando scatta l'obbligo di notifica previsto dall'art. 33 GDPR? *Traccia.* Quando il titolare viene a conoscenza di una violazione di dati personali, salvo che sia improbabile che essa presenti un rischio per i diritti e le libertà delle persone fisiche: la notifica va fatta senza ingiustificato ritardo e, ove possibile, entro 72 ore. Se il rischio è elevato si aggiunge la comunicazione agli interessati (art. 34).

5. Che differenza c'è tra playbook e runbook? *Traccia.* Il playbook definisce strategia, decisioni e responsabilità per una categoria di incidenti e cambia con l'organizzazione e le norme. Il runbook descrive la procedura operativa passo per passo, segue l'evoluzione della tecnologia ed è la parte che si presta più direttamente all'automazione.

Capitolo 2. Introduzione all'AI per la sicurezza

Il capitolo fornisce i fondamenti di machine learning e di deep learning necessari a valutare le soluzioni di sicurezza basate sull'intelligenza artificiale, con un lessico che i capitoli successivi danno per acquisito. Presenta i paradigmi di apprendimento e la pipeline con cui un modello viene costruito e mantenuto, i modelli linguistici di grandi dimensioni (LLM) e il loro impiego come agenti, i limiti statistici che rendono la detection un problema diverso dalla maggior parte delle applicazioni del machine learning. Gli approfondimenti trattano le metriche di valutazione, il concept drift, la scelta del paradigma, il feature engineering e la base rate fallacy.

Percorso. Traccia core, in apertura del corso.

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- distinguere apprendimento supervisionato, non supervisionato e per rinforzo, indicando per ciascuno un uso in sicurezza;
- interpretare precision, recall, F1 e curve ROC nel contesto della detection;
- discutere i limiti statistici dei classificatori applicati a eventi rari e stimarne l'effetto sul numero di falsi positivi.

2.1 Paradigmi di apprendimento e usi in sicurezza

Il machine learning studia algoritmi che migliorano le proprie prestazioni in un compito a partire dai dati, senza che le regole di decisione siano scritte esplicitamente.

Nell'**apprendimento supervisionato** il modello riceve esempi già etichettati, per esempio file classificati come malevoli o benigni, e impara una funzione che associa le caratteristiche dell'esempio all'etichetta. È il paradigma dei classificatori di malware e di phishing. La sua qualità dipende dalla disponibilità di etichette corrette, aggiornate e rappresentative dei dati che il modello incontrerà in esercizio.

Nell'**apprendimento non supervisionato** le etichette mancano. Il modello cerca strutture nei dati: gruppi di elementi simili (clustering) o elementi che si discostano dalla norma appresa (anomaly detection). In sicurezza è alla base dei sistemi di analisi comportamentale (UEBA), che costruiscono una baseline per utenti e dispositivi e segnalano le deviazioni. Quando la baseline è costruita su un periodo ritenuto privo di attacchi, la tecnica rientra nell'anomaly detection semi-supervisionata, in cui i dati di addestramento contengono solo esempi normali (Chandola, Banerjee e Kumar, 2009). Questi sistemi possono rilevare comportamenti mai osservati, ma un'anomalia non coincide con un attacco e il numero di falsi positivi tende a essere elevato.

Nell'apprendimento per rinforzo un agente interagisce con un ambiente, riceve una ricompensa per le proprie azioni e ne ricava una politica. In sicurezza è studiato per la scelta automatica delle azioni di risposta e per la gestione di honeypot adattivi; le applicazioni in produzione

restano limitate, soprattutto perché l'apprendimento per tentativi su sistemi in esercizio comporta rischi difficili da accettare.

Qualunque sia il paradigma, la costruzione di un modello segue una pipeline: raccolta dei dati, estrazione delle feature, addestramento, validazione, rilascio e monitoraggio. L'**overfitting** si verifica quando il modello si adatta ai dettagli del campione di addestramento e generalizza male; si controlla con insiemi di validazione separati e tecniche come la validazione incrociata. Il concept drift, trattato nel § 2.5, degrada invece il modello dopo il rilascio. In sicurezza la separazione tra dati di addestramento e di test va fatta anche nel tempo: un modello va valutato su dati successivi a quelli di addestramento, altrimenti le prestazioni misurate sono sovrastimate.

2.2 Addestramento e impiego degli LLM

I modelli linguistici di grandi dimensioni sono reti neurali basate sull'architettura transformer, introdotta nel 2017, addestrate a prevedere la parte successiva di un testo su corpora molto ampi. Questa fase, il **pre-training**, produce un modello con conoscenze generali e capacità linguistiche, ma privo di conoscenze specifiche sull'organizzazione che lo usa e con informazioni aggiornate solo fino alla data di raccolta del corpus.

Il **fine-tuning** adatta un LLM a un dominio modificandone i pesi con un ulteriore addestramento su dati specifici; è utile per imporre un formato di output o un lessico di settore, ma richiede dati curati e va ripetuto quando le conoscenze cambiano. La retrieval-augmented generation (RAG) lascia invariati i pesi: a ogni richiesta un sistema di ricerca recupera i documenti pertinenti (playbook, report di incidenti, documentazione tecnica) e li inserisce nel contesto del modello, che risponde sulla base di quei testi e può citarli. Il RAG è aggiornabile senza riaddestramento e, se il sistema riporta i passaggi recuperati, rende la risposta verificabile sulle fonti; il modello può tuttavia travisare o ignorare i documenti forniti (capitolo 14).

Un **agente** è un LLM collegato a strumenti esterni, come la ricerca su un SIEM, la consultazione di fonti di threat intelligence o l'apertura di un ticket, e inserito in un ciclo in cui pianifica, esegue un'azione e ne valuta il risultato. Con l'uso di strumenti il modello compie azioni sui sistemi: occorrono limiti di privilegio, registrazione di ogni chiamata e approvazione umana per le operazioni che modificano lo stato dei sistemi (capitolo 6).

In tutti questi impieghi la spiegabilità (XAI, explainable AI) è un requisito: in contesti critici l'output di un modello va accompagnato da elementi con cui un analista possa verificarlo. Il NIST AI Risk Management Framework (AI RMF 1.0, 2023) organizza la gestione del rischio dei sistemi di AI in quattro funzioni, Govern, Map, Measure e Manage, e costituisce un riferimento per valutarne l'adozione.

2.3 Limiti statistici della detection

La detection degli attacchi presenta caratteristiche che la distinguono dalla maggior parte dei problemi di classificazione. Le classi sono fortemente sbilanciate: gli eventi malevoli sono una frazione minima del totale. In queste condizioni l'accuratezza, cioè la quota di decisioni cor-

rette, è poco informativa. Un sistema con accuratezza del 99%, applicato a dati in cui gli attacchi sono un evento su 100.000, sbaglia circa un evento legittimo ogni cento e produce migliaia di falsi allarmi al giorno anche con volumi di traffico modesti.

Per questo si usano metriche che separano i due tipi di errore, come precision, recall e F1 (§ 2.4). La soglia di allarme diventa una scelta di costo tra falsi positivi, che consumano il tempo degli analisti, e falsi negativi, che lasciano attacchi non rilevati. Lo sbilanciamento va trattato sia in addestramento, con tecniche di campionamento o di pesatura delle classi, sia in valutazione, misurando le prestazioni con una prevalenza realistica.

Sommer e Paxson, in un articolo presentato all'IEEE Symposium on Security and Privacy del 2010, hanno spiegato perché l'anomaly detection basata sul machine learning, molto studiata in ambito accademico, trovi scarso impiego operativo. Le ragioni indicate, utili anche per valutare i prodotti commerciali, sono la tendenza del machine learning a riconoscere le somiglianze con i dati di addestramento meglio delle attività estranee (assunzione del mondo chiuso), il costo elevato degli errori, la distanza semantica tra un'anomalia statistica e un attacco, la variabilità del traffico di rete, che rende instabile la nozione di normalità, e la difficoltà di valutazione dovuta alla mancanza di dataset realistici e al contesto avversario.

2.4 Precision, recall e soglia di allarme

Data una matrice di confusione con veri positivi (VP), falsi positivi (FP), falsi negativi (FN) e veri negativi (VN), la **precision** è $VP/(VP+FP)$: la quota di alert fondati tra quelli emessi. Il **recall** è $VP/(VP+FN)$: la quota di attacchi presenti che il sistema rileva. La F1 è la media armonica delle due, $2 \cdot P \cdot R / (P + R)$, e riassume il bilanciamento in un solo valore.

Esempio. Su 100 attacchi presenti il modello ne segnala 90, con recall 0,90, e produce anche 60 falsi allarmi: la precision è $90/150 = 0,60$ e la F1 vale 0,72.

Al variare della soglia di decisione le due metriche tendono a muoversi in direzioni opposte: una soglia più alta riduce o lascia invariati i falsi positivi e di norma aumenta la precision, anche se non in modo monotono, ma può lasciar passare più attacchi, per cui il recall diminuisce o resta invariato. La curva ROC riporta il tasso di veri positivi in funzione del tasso di falsi positivi per tutte le soglie; l'area sottesa (AUC) misura la capacità del modello di separare le classi indipendentemente dalla soglia e serve a confrontare modelli diversi. In esercizio, però, il sistema opera con una sola soglia, scelta in base ai costi. Con classi molto sbilanciate la curva precision-recall è più informativa della ROC, perché il tasso di falsi positivi può restare basso in percentuale anche quando i falsi positivi sono numerosi in valore assoluto (Saito e Rehmsmeier, 2015).

2.5 Concept drift naturale e avversario

Un modello apprende la distribuzione statistica dei dati disponibile al momento dell'addestramento. Quando questa distribuzione cambia, le prestazioni calano: il fenomeno è detto **concept drift**. Il drift naturale nasce da cambiamenti legittimi dell'ambiente: nuove applicazioni, migra-

zioni verso il cloud, lavoro da remoto, cambi di abitudini degli utenti. Il **drift avversario** è prodotto dagli attaccanti, che osservano quali comportamenti vengono rilevati e modificano le proprie tecniche per eludere il modello. Questa componente distingue il machine learning applicato alla sicurezza dalla maggior parte degli altri domini, in cui i dati non cambiano per effetto di un avversario.

Esempio. Un classificatore di phishing addestrato prima della diffusione dei modelli linguistici generativi si basa in parte su errori grammaticali e costrutti innaturali. Nei messaggi prodotti con un LLM queste caratteristiche scompaiono e le prestazioni del classificatore calano (capitolo 12).

Le contromisure comprendono il monitoraggio della distribuzione degli input e delle prestazioni in esercizio, il riaddestramento periodico con dati recenti e allarmi quando il tasso di segnalazione si discosta dalla serie storica. La ricerca ha proposto metodi specifici per rilevare il drift nei classificatori di malware (Jordaney et al., 2017) e protocolli di valutazione che rispettano l'ordine temporale dei dati (Pendlebury et al., 2019). Il drift avversario è ripreso nel capitolo 13, dove l'attaccante sfrutta deliberatamente i meccanismi di aggiornamento delle baseline.

2.6 Agenti basati su LLM

Un LLM usato da solo trasforma testo in testo, e il suo errore possibile è una risposta sbagliata, che un lettore può riconoscere e scartare. Un agente opera in un ciclo: osserva lo stato, pianifica il passo successivo, invoca uno strumento, valuta il risultato e decide come proseguire. Gli strumenti hanno effetti sui sistemi. Un errore può quindi tradursi in un'azione, come la chiusura di un caso o il blocco di un account, e la sequenza di passi lo amplifica: una premessa errata al primo passo orienta tutti i successivi.

Scenario didattico. Un agente di triage legge un alert, interroga il SIEM sull'host coinvolto, consulta la threat intelligence sull'indirizzo IP di destinazione e propone di chiudere il caso come falso positivo. Le tre chiamate agli strumenti, i loro risultati e la conclusione vanno registrati, perché un analista possa verificare il ragionamento e individuare l'eventuale passo errato.

Ogni strumento concesso all'agente è una capacità operativa. Si applicano il principio del minimo privilegio, l'accesso in sola lettura come impostazione predefinita e l'approvazione umana per le operazioni di scrittura. Questi principi sono sviluppati nei capitoli 6 (agenti nel SOC), 13 (attacchi ai sistemi di AI, in particolare la prompt injection) e 14 (progettazione e osservabilità dei sistemi difensivi).

2.7 Scelta del paradigma di apprendimento

La scelta del paradigma dipende in primo luogo dalle etichette disponibili. Dove esistono grandi raccolte etichettate, come nel caso di malware e phishing, l'apprendimento supervisionato offre prestazioni elevate sui fenomeni noti. Dove le etichette mancano, o dove interessa rilevare comportamenti nuovi, si ricorre all'apprendimento non supervisionato, accettando un numero maggiore di falsi positivi; l'apprendimento per rinforzo resta invece confinato soprattutto alla ricerca e alla simulazione.

Esempio. Per individuare account compromessi, un modello supervisionato addestrato sugli incidenti passati riconosce schemi già osservati, mentre un modello non supervisionato sul comportamento degli account (orari, volumi, destinazioni) segnala l'account che si discosta dal proprio storico anche in presenza di tecniche nuove. Le due soluzioni si possono combinare in sequenza: il modello non supervisionato produce le segnalazioni e quello supervisionato aiuta a ordinarle.

Le etichette dei dati di sicurezza sono sbilanciate, perché gli attacchi sono rari, e imperfette, perché gli attacchi etichettati sono quelli scoperti, mentre i dati considerati benigni possono contenere attacchi passati inosservati. Un modello supervisionato addestrato su questi dati eredita le lacune della detection precedente e tende a considerare normali i comportamenti che nessuno aveva rilevato.

2.8 Feature engineering sui log di sicurezza

Un modello di machine learning opera su vettori numerici. Il **feature engineering** traduce i dati grezzi, come i log, in questa forma. Da un flusso di eventi di autenticazione si possono estrarre, per esempio:

- il numero di accessi falliti per finestra temporale;
- il rapporto tra accessi riusciti e falliti;
- il numero di host distinti contattati da un account;
- l'ora del giorno, codificata come variabile ciclica (seno e coseno dell'angolo orario) perché le 23 e l'una risultino vicine;
- la novità della coppia utente-dispositivo rispetto allo storico.

Un modello semplice con feature che descrivono il comportamento d'attacco ottiene spesso risultati migliori di un modello complesso con feature povere.

Esempio. Nel password spraying l'attaccante prova poche password comuni su molti account. Il conteggio degli accessi falliti per account non evidenzia anomalie, perché ogni account riceve uno o due tentativi. L'attacco emerge aggregando per origine: il numero di account distinti con accessi falliti provenienti dallo stesso indirizzo IP in una finestra temporale; se l'attacco è distribuito su molti indirizzi servono aggregazioni sull'intero tenant, come il numero complessivo di account con accessi falliti nella finestra.

Il deep learning può apprendere rappresentazioni direttamente da sequenze di eventi (embedding), riducendo il lavoro manuale, con una minore spiegabilità e una maggiore dipendenza dalla quantità di dati. Quando il risultato va difeso, come in ambito forense o disciplinare, le feature esplicite e documentate sono la scelta prudente: ogni feature è una scelta progettuale che deve poter essere spiegata (capitolo 8).

2.9 Base rate fallacy e falsi positivi

La base rate fallacy è l'errore di giudizio che consiste nel trascurare la frequenza di base di un fenomeno quando si interpreta un risultato. Il teorema di Bayes fornisce la probabilità che un alert corrisponda a un attacco, detta valore predittivo positivo: $VPP = (R \cdot p) / (R \cdot p + FPR \cdot (1 - p))$, dove R è il recall, FPR il tasso di falsi positivi e p la prevalenza degli attacchi. Stefan Axelsson ha mostrato nel 2000 che questo vincolo rende la detection intrinsecamente difficile: con prevalenze molto basse servono tassi di falsi positivi estremamente ridotti per ottenere alert affidabili.

Scenario didattico. Si consideri un detector con recall del 99% e tasso di falsi positivi dell'1%, applicato a 10 milioni di eventi al giorno di cui 100 malevoli. Produce circa 99 veri positivi e circa 100.000 falsi positivi: la probabilità che un alert corrisponda a un attacco è circa 99 su 100.100, cioè lo 0,1%. Ridurre il tasso di falsi positivi allo 0,01% porterebbe i falsi positivi a circa 1.000 e il valore predittivo positivo a circa il 9%.

Le affermazioni dei fornitori vanno lette con questo schema. Una percentuale di attacchi rilevati e un tasso di falsi positivi misurano grandezze diverse e nessuna delle due indica da sola quanti alert al giorno riceverà un SOC: in una prova sul campo occorre misurare il numero di alert giornalieri sul traffico dell'organizzazione e la quota di quelli fondati.

Un singolo detector difficilmente sostiene questi volumi. Si progettano quindi architetture a cascata: un primo stadio ad alto recall seleziona i candidati, stadi successivi (arricchimento con informazioni di contesto, correlazione tra fonti, un secondo modello più costoso) li filtrano prima che arrivino all'analista. Questa architettura è ripresa nei capitoli 3 e 5.

2.10 Laboratorio: classificatore su traffico di rete

Il classificatore si costruisce in Python, con la libreria scikit-learn, sul dataset pubblico CIC-IDS2017, prodotto dal Canadian Institute for Cybersecurity, che contiene traffico di rete benigno e traffico generato da diversi tipi di attacco, con feature calcolate per flusso. Il dataset è molto usato in letteratura, ma presenta problemi noti di generazione del traffico, costruzione dei flussi, estrazione delle feature ed etichettatura (Engelen et al., 2021); gli autori ne hanno pubblicato una versione corretta, da preferire nel laboratorio.

Dopo un'analisi esplorativa del dataset si addestra un classificatore che distingue traffico benigno e malevolo e lo si valuta con precision, recall, F1, curva ROC e curva precision-recall. Si simula poi una prevalenza realistica degli attacchi, ricampionando il traffico malevolo, e se ne misura l'effetto sul numero di falsi positivi giornalieri.

Consegna: notebook con le metriche commentate; stima dei falsi positivi al giorno con una prevalenza realistica; una pagina di valutazione sull'impiego del modello in un SOC. L'obiettivo è mostrare la distanza tra le metriche ottenute sul dataset e le prestazioni attese in esercizio.

2.11 Letture

- I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016, capitoli 1 e 5.
- V. Chandola, A. Banerjee, V. Kumar, «Anomaly Detection: A Survey», *ACM Computing Surveys*, 41(3), 2009.
- NIST AI 100-1, *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*, 2023.
- R. Sommer, V. Paxson, «Outside the Closed World: On Using Machine Learning for Network Intrusion Detection», *IEEE Symposium on Security and Privacy*, 2010.
- T. Saito, M. Rehmsmeier, «The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets», *PLoS ONE*, 10(3), e0118432, 2015.
- R. Jordaney et al., «Transcend: Detecting Concept Drift in Malware Classification Models», *USENIX Security Symposium*, 2017.
- F. Pendlebury et al., «TESSERACT: Eliminating Experimental Bias in Malware Classification across Space and Time», *USENIX Security Symposium*, 2019.
- S. Axelsson, «The Base-Rate Fallacy and the Difficulty of Intrusion Detection», *ACM Transactions on Information and System Security*, 3(3), 2000.
- I. Sharafaldin, A. H. Lashkari, A. A. Ghorbani, «Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization», *ICISSP*, 2018.
- G. Engelen, V. Rimmer, W. Joosen, «Troubleshooting an Intrusion Detection Dataset: the CICIDS2017 Case Study», *IEEE Security and Privacy Workshops*, 2021.

2.12 Domande di autovalutazione

1. Perché un'accuratezza del 99% può essere insufficiente nella detection? *Traccia.* Con classi fortemente sbilanciate l'accuratezza è dominata dagli eventi legittimi: un sistema accurato al 99% sbaglia circa un evento legittimo ogni cento e, con prevalenze di un attacco ogni 100.000 eventi, produce falsi allarmi in numero molto superiore agli attacchi rilevati. Il valore predittivo positivo, calcolato con il teorema di Bayes, resta molto basso.
2. Cosa distingue fine-tuning e RAG? *Traccia.* Il fine-tuning modifica i pesi del modello con dati di dominio e va ripetuto quando le conoscenze cambiano; il RAG lascia invariato il modello e recupera a ogni richiesta i documenti pertinenti, rendendo la risposta aggiornabile e, quando il sistema cita i passaggi usati, verificabile sulle fonti.
3. Cos'è il concept drift e perché degrada i modelli di sicurezza? *Traccia.* È il cambiamento nel tempo della distribuzione dei dati rispetto a quella di addestramento. In sicurezza ha una componente naturale (cambiamenti dell'ambiente) e una avversaria (attaccanti che adattano le tecniche); richiede monitoraggio delle prestazioni e riaddestramento periodico.

4. Quali metriche privilegiare con classi fortemente sbilanciate? *Traccia.* Precision, recall, F1 e curva precision-recall, lette insieme ai volumi assoluti di alert attesi sul traffico di produzione dell'organizzazione. L'accuratezza e, in parte, la curva ROC danno un quadro ottimistico quando la prevalenza è molto bassa.
5. Cosa si intende per agente autonomo basato su LLM? *Traccia.* Un modello linguistico inserito in un ciclo di pianificazione ed esecuzione che invoca strumenti esterni con effetti sui sistemi. Richiede minimo privilegio, registrazione di ogni chiamata e approvazione umana per le azioni che modificano lo stato dei sistemi.

Capitolo 3. Detection potenziata dall'AI

Il capitolo esamina la fase di rilevamento e il contributo che vi portano le tecniche di machine learning, alla luce dei limiti statistici introdotti nel capitolo 2. Confronta la detection basata su firme, su regole e su anomalie, descrive i sistemi di analisi comportamentale (UEBA) e il triage automatizzato degli alert, e presenta l'architettura di una pipeline di detection in cui componenti tradizionali e modelli statistici convivono. Gli approfondimenti trattano la costruzione delle baseline, il calcolo del punteggio di rischio, la misura della copertura con MITRE ATT&CK, il rapporto tra regole e modelli, l>alert fatigue e il detection engineering.

Percorso. Specializzazione Detection & Response (capitoli 3–6).

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- confrontare la detection basata su firme, su regole e su anomalie, indicando punti di forza e limiti di ciascun approccio;
- valutare un sistema UEBA in termini di copertura delle tecniche MITRE ATT&CK;
- progettare una pipeline di triage degli alert che includa componenti di machine learning.

3.1 Firme, regole e anomalie

La detection basata su **firme** confronta i dati osservati con indicatori noti: valori di hash di file malevoli, indirizzi e domini associati a infrastrutture d'attacco, sequenze di byte descritte da regole YARA, pattern di traffico riconosciuti da sistemi come Snort o Suricata. È precisa sulle minacce già catalogate e produce pochi falsi positivi, ma non rileva varianti nuove o polimorfiche, che modificano gli elementi su cui la firma si basa.

La detection basata su **regole** descrive esplicitamente comportamenti: una sequenza di eventi, una combinazione di processo padre e processo figlio, un accesso a una risorsa in un orario non previsto. Nel formato aperto Sigma, mantenuto dal progetto SigmaHQ, si scrivono regole indipendenti dal SIEM, convertibili nei linguaggi di interrogazione dei diversi prodotti. Le regole sono trasparenti e verificabili; richiedono manutenzione e possono essere eluse da un attaccante che conosce la logica di detection.

La detection basata su anomalie usa modelli statistici o di machine learning per stimare un comportamento normale e segnalare gli scostamenti. Chandola, Banerjee e Kumar, nella rassegna pubblicata su ACM Computing Surveys nel 2009, distinguono anomalie puntuali (un singolo valore fuori norma), contestuali (un valore anomalo in un certo contesto, per esempio un accesso legittimo di giorno e sospetto di notte) e collettive (una sequenza di eventi anomala nel suo insieme). L'approccio rileva novità e abusi da parte di utenti interni, al prezzo di falsi positivi più frequenti e di una spiegabilità inferiore.

I tre approcci sono complementari. La copertura complessiva si valuta mappando le detection sulle tecniche descritte da MITRE ATT&CK, la base di conoscenza delle tattiche e tecniche osservate in attacchi documentati, e le contromisure sul catalogo MITRE D3FEND, che descrive le

tecniche difensive e le collega alle tecniche offensive di ATT&CK.

3.2 UEBA e triage automatizzato

I sistemi di User and Entity Behavior Analytics (**UEBA**) costruiscono una baseline per ogni utente ed entità (host, account di servizio, applicazione) sulla base di orari, host contattati, volumi di dati, aree geografiche di accesso e modalità di autenticazione. Per misurare lo scarto dalla norma usano tecniche statistiche, algoritmi di clustering e modelli come gli autoencoder, reti neurali che imparano a ricostruire i dati normali e segnalano come anomali gli elementi ricostruiti con errore elevato. Gli scarti si traducono in un punteggio di rischio cumulativo per entità (§ 3.5). I sistemi UEBA sono efficaci soprattutto nel rilevare abusi interni e account compromessi, casi in cui l'attaccante usa credenziali valide e non attiva le firme.

Il machine learning si applica anche al triage. Con volumi elevati di alert la prioritizzazione diventa il collo di bottiglia del SOC: gli strumenti di machine learning deduplicano gli alert ripetuti, correlano quelli che riguardano la stessa entità e li ordinano per rischio. Le principali piattaforme SIEM e XDR offrono funzioni di questo tipo, come il motore di correlazione degli incidenti di Microsoft Defender XDR, che nel portale Defender sostituisce la regola Fusion di Microsoft Sentinel (il portale Azure non supporterà più Sentinel dopo il 31 marzo 2027), la funzione Attack Discovery di Elastic Security e il risk-based alerting di Splunk Enterprise Security. I job di anomaly detection di Elastic, disponibili con licenza a pagamento, e lo Splunk AI Toolkit (già Machine Learning Toolkit) servono invece alla detection basata su anomalie.

Il rischio specifico del triage automatico è la produzione di falsi negativi silenziosi: un alert classificato come irrilevante e chiuso senza revisione non lascia traccia nel lavoro degli analisti. Per questo occorre misurare anche ciò che il sistema scarta, per esempio con revisioni a campione degli alert chiusi automaticamente. Un punteggio di rischio non spiegabile pone inoltre problemi giuridici quando incide su persone (capitolo 15).

3.3 La pipeline di detection assistita

Una pipeline di detection assistita si articola in quattro stadi. La **telemetria** raccoglie dati da endpoint, rete, sistemi di identità e servizi cloud. Lo stadio di detection applica firme, regole e modelli. Lo stadio di correlazione deduplica gli alert, li arricchisce con informazioni di contesto (criticità dell'asset, ruolo dell'utente, reputazione degli indicatori) e calcola un punteggio di rischio. Lo stadio di triage presenta all'analista gli alert ordinati con il contesto necessario alla decisione. In questo capitolo l'ultimo stadio resta affidato all'analista; l'automazione della risposta è trattata nel capitolo 5.

La qualità della pipeline dipende dalla telemetria: un modello sofisticato non compensa l'assenza di una fonte di log. La sua maturità si misura con la copertura delle tecniche ATT&CK pesata per la rilevanza nel proprio threat model, cioè nel modello delle minacce plausibili per l'organizzazione, e non con il numero di regole attive, spesso usato nei rapporti interni.

3.4 Costruzione di una baseline comportamentale

La **baseline** è il modello di normalità rispetto al quale si misura lo scarto. Si costruisce per entità e su una finestra storica abbastanza lunga da includere la stagionalità: settimane o mesi anziché giorni, perché l'attività cambia tra giorni feriali e fine settimana e nei periodi di chiusura contabile. Le dimensioni tipiche sono gli orari di attività, gli host e le aree geografiche di accesso, i volumi di dati trasferiti, le applicazioni usate e le modalità di autenticazione.

Scenario didattico. Si supponga che un amministratore, che di norma accede dalle 8 alle 18 da due postazioni, compaia alle 3 di notte, da una postazione diversa da quelle abituali, su un server che non aveva usato in precedenza. Lo scarto riguarda tre dimensioni insieme (orario, host di origine, host di destinazione), e la combinazione di anomalie è un indizio più forte di ciascuna anomalia isolata.

Se l'attaccante è già presente durante la fase di apprendimento, il suo comportamento entra nella baseline come normalità, un caso di avvelenamento dei dati trattato nel capitolo 13. La baseline va inoltre aggiornata dopo ogni riorganizzazione, perché un cambio di ruolo legittimo produce scarti simili a quelli di un account compromesso. Le procedure di gestione del personale (cambi di mansione, nuovi incarichi) dovrebbero quindi alimentare il sistema di analisi comportamentale. Quando riguarda i dipendenti, l'analisi comportamentale ricade nella disciplina dei controlli a distanza (art. 4 della L. 20 maggio 1970, n. 300) e richiede una valutazione d'impatto ai sensi dell'art. 35 del Reg. (UE) 2016/679 (capitolo 15).

3.5 Composizione e limiti del punteggio di rischio

In molti sistemi UEBA il punteggio di rischio è una somma pesata nel tempo. Ogni anomalia aggiunge punti in proporzione alla sua rarità e alla criticità dell'entità coinvolta; a seconda del prodotto, i contributi decadono progressivamente o escono da una finestra di osservazione se non seguono altri segnali. Il meccanismo intercetta le campagne lente e a basso profilo, in cui l'attaccante compie azioni singolarmente poco rilevanti distribuite nel tempo, che sfuggono a una regola applicata al singolo evento ma, sommate sulla stessa entità, superano la soglia.

Scenario didattico. Un accesso da una VPN insolita vale 15 punti, il primo accesso a un file server del personale 20, il download di 2 GB fuori orario 35, la creazione di un archivio cifrato 25. Il totale di 95 punti supera la soglia di 80 e l'alert descrive una possibile esfiltrazione in corso.

Un punteggio presentato senza la scomposizione nei suoi contributi non è verificabile né contestabile: l'analista deve vedere gli eventi che lo compongono e i pesi applicati. Occorre inoltre chiedersi quali comportamenti legittimi produrrebbero lo stesso punteggio, per esempio un backup manuale o il trasferimento dei dati di un progetto. Quando il punteggio incide su decisioni che riguardano persone, come un provvedimento disciplinare, la trasparenza diventa anche un requisito giuridico (capitolo 15).

3.6 Copertura ATT&CK come misura della detection

La capacità di rilevamento si misura con la quota di tecniche rilevanti per il proprio threat model coperte da almeno una detection. Si individuano gli attori e le campagne plausibili per il proprio settore, con il supporto della threat intelligence (capitolo 4), e se ne estraggono le tecniche ATT&CK. Ogni detection esistente viene poi associata alla tecnica che rileva e il risultato si rappresenta sulla matrice, per esempio con lo strumento ATT&CK Navigator o con il framework DeTT&CT; le tecniche rimaste scoperte indicano le priorità del detection engineering.

Scenario didattico. La tecnica T1566 (phishing) è coperta dal gateway di posta e dalla sandbox, la T1003 (OS credential dumping) dall'EDR. La T1021.001 (Remote Desktop Protocol, sotto-tecnica di T1021 Remote Services) è invece scoperta, perché manca visibilità sul movimento laterale tramite RDP interno, e diventa l'obiettivo del successivo ciclo di lavoro.

Una regola fragile basata su una sola fonte di log non equivale a tre segnali indipendenti, e una detection mai verificata può non funzionare. Qualificare la copertura per fonte di telemetria e verificarla con test di attacco simulato, per esempio con i test della libreria Atomic Red Team, evita una rappresentazione ottimistica della matrice.

3.7 Regole e modelli nella stessa architettura

Regole e modelli hanno proprietà complementari. Una regola è precisa, spiegabile, veloce e versionabile, ma rileva solo ciò che il suo autore ha previsto. Un modello generalizza e può segnalare comportamenti imprevisti, con più falsi positivi, maggiore bisogno di dati e minore spiegabilità. Una detection matura li combina: regole per i comportamenti noti e per i vincoli di conformità (per esempio qualsiasi accesso a un database critico fuori orario), modelli per le anomalie e per ordinare per priorità l'output delle regole.

Esempio. Per rilevare l'uso del comando certutil con l'opzione -urlcache per scaricare file, una tecnica che sfrutta un binario legittimo di Windows, basta una regola Sigma, senza componenti di machine learning. Un movimento laterale lento eseguito con credenziali valide non è descritto da alcuna regola e può essere segnalato da un modello UEBA.

Gli LLM producono bozze accettabili di regole Sigma a partire dalla descrizione di una tecnica (capitolo 6). Una regola generata va validata su log storici, misurandone i falsi positivi, prima del rilascio.

3.8 Alert fatigue e risk-based alerting

L'**alert fatigue** è l'effetto sugli analisti dell'elevato numero di falsi positivi prodotto dalla bassa prevalenza degli attacchi (§ 2.9): chi chiude ogni giorno centinaia di falsi positivi riduce progressivamente l'attenzione dedicata a ciascun alert. I segnali sono tempi di presa in carico crescenti, chiusure in blocco come benigni e regole disattivate per l'eccesso di rumore.

Caso documentato. Nella violazione subita dalla catena statunitense Target nel 2013 il sistema di rilevamento aveva generato alert relativi all'installazione del malware, che non furono gestiti in tempo. Il fatto è riportato nel rapporto dello staff di maggioranza della commissione Commercio, Scienza e Trasporti del Senato degli Stati Uniti (26 marzo 2014), che ipotizza, senza accertarlo, che gli alert siano stati scambiati per falsi positivi di un sistema che segnalava con frequenza.

Il **risk-based alerting** affronta il problema modificando l'unità di lavoro: invece di inviare all'analista ogni segnale, il sistema accumula punteggi di rischio per entità e genera un alert quando la somma supera una soglia. Molti segnali deboli sullo stesso host diventano un unico alert con il contesto aggregato. Ogni regola e ogni modello hanno inoltre un costo in attenzione, stimabile come prodotto tra numero di alert generati e tempo medio di gestione, da confrontare con la quota di alert fondati; una revisione periodica che corregge o disattiva le fonti più rumorose libera tempo per gli alert rilevanti. Il rapporto tra fiducia nel sistema e controllo umano è ripreso nel capitolo 5 a proposito dell'automation bias.

3.9 Detection engineering e detection-as-code

Il **detection engineering** tratta le detection come software sottoposto a un ciclo di sviluppo. Si parte da una minaccia prioritaria, individuata con la threat intelligence e mappata su ATT&CK. Si scrive la detection come codice versionato, secondo la pratica detta detection-as-code: repository, revisione da parte di un secondo analista, integrazione continua con test automatici. La si verifica con attacchi simulati, con test di Atomic Red Team o con esercitazioni purple team, in cui attaccanti e difensori lavorano insieme. Si misurano falsi positivi e copertura, si rilascia la detection e se ne monitora il decadimento, perché ogni detection perde efficacia con il cambiare dell'avversario e dell'ambiente.

Scenario didattico. Un ciclo di lavoro riguarda la tecnica T1055 (process injection). La regola è scritta e salvata nel repository con la documentazione, e un test automatico esegue il corrispondente test di Atomic Red Team in un ambiente di staging. La soglia è tarata su due settimane di log storici, poi la regola viene rilasciata. Sei mesi dopo il test periodico fallisce perché un aggiornamento dell'EDR ha cambiato il formato dei log, e segnala un guasto della detection che altrimenti sarebbe passato inosservato.

La mappa di copertura registra per ciascuna detection la data dell'ultima verifica. L'AI può accelerare ogni fase del ciclo, dalle bozze di regole alla generazione di varianti di test e al triage del backlog; restano decisioni umane la scelta delle minacce prioritarie e il giudizio su quando una detection è adeguata, perché dipendono dalla valutazione del rischio dell'organizzazione.

3.10 Laboratorio: anomaly detection e regole Sigma a confronto

L'ambiente di lavoro è Security Onion, una piattaforma gratuita per il monitoraggio della sicurezza che integra lo stack Elastic (i componenti di Security Onion ed Elastic sono distribuiti con la Elastic License 2.0), installata in un ambiente isolato predisposto dal docente. Le funzioni di

anomaly detection del machine learning di Elastic non rientrano nella licenza gratuita di base: si può usare una licenza di prova oppure un modello equivalente in Python, per esempio Isolation Forest di scikit-learn, applicato ai log esportati.

Si crea un job di anomaly detection sui log di autenticazione e si scrivono le regole Sigma equivalenti per gli stessi scenari. Gli attacchi simulati si generano con Atomic Red Team, eseguito solo nell'ambiente di laboratorio, e sugli stessi dati si misurano precision e recall dei due approcci.

Consegna: tabella di precision e recall per approccio; mappatura delle detection sulle tecniche ATT&CK; analisi dei casi in cui il machine learning ha aggiunto valore e di quelli in cui ha aggiunto rumore. L'attenzione va alla misura comparata, indipendentemente dallo strumento usato.

3.11 Letture

- V. Chandola, A. Banerjee, V. Kumar, «Anomaly Detection: A Survey», *ACM Computing Surveys*, 41(3), 2009.
- MITRE, *ATT&CK Enterprise Matrix* e documentazione delle tecniche (attack.mitre.org).
- MITRE, *D3FEND*, base di conoscenza delle contromisure difensive (d3fend.mitre.org).
- SigmaHQ, specifica del formato Sigma e repository delle regole.
- ENISA, *Threat Landscape*, edizione più recente.
- U.S. Senate Committee on Commerce, Science and Transportation, *A «Kill Chain» Analysis of the 2013 Target Data Breach*, Majority Staff Report for Chairman Rockefeller, 26 marzo 2014.

3.12 Domande di autovalutazione

1. Quali attacchi sfuggono alla detection basata su firme? *Traccia.* Le varianti nuove o polimorfiche, gli attacchi che usano strumenti legittimi del sistema e quelli condotti con credenziali valide: in tutti questi casi mancano gli elementi fissi (hash, indicatori, sequenze di byte) su cui la firma si basa.
2. Come si costruisce una baseline comportamentale affidabile? *Traccia.* Per entità, su una finestra storica che includa la stagionalità, con dimensioni scelte in base al comportamento da descrivere. Va verificata l'assenza di attività malevole nel periodo di apprendimento e va aggiornata dopo le riorganizzazioni.
3. Quali rischi comporta un punteggio di rischio opaco? *Traccia.* L'analista non può verificarlo né confrontarlo con spiegazioni legittime, e gli errori del modello passano inosservati. Quando incide su persone, l'opacità impedisce di motivare e contestare la decisione, con conseguenze anche giuridiche.
4. In che modo il triage automatico può introdurre falsi negativi? *Traccia.* Chiudendo o declassando alert che corrispondono ad attacchi effettivi: questi errori non entrano nel lavoro degli analisti e restano invisibili. Si mitigano con revisioni a campione degli alert scartati e con metriche che misurano anche le chiusure automatiche.

5. Come si misura la copertura ATT&CK di un set di regole? *Traccia.* Si individuano le tecniche rilevanti per il proprio threat model, si associa ogni regola alla tecnica che rileva e si rappresenta il risultato sulla matrice, qualificando la copertura per fonte di telemetria e verificandola con test di attacco simulato.

Capitolo 4. Threat Intelligence e AI

Il capitolo tratta la threat intelligence, cioè la produzione di conoscenza sulle minacce utile alle decisioni di difesa, e il contributo che vi danno le tecniche di elaborazione del linguaggio naturale (NLP) e i modelli linguistici. Descrive il ciclo dell'intelligence e i suoi livelli, l'estrazione automatica di indicatori e tecniche da fonti testuali, la valutazione dell'attendibilità delle fonti. Gli approfondimenti riguardano la piramide del dolore, gli standard di condivisione STIX, TAXII e TLP, gli errori tipici dell'estrazione automatica, i destinatari dei diversi prodotti, l'attribuzione e le comunità di condivisione. L'intelligence alimenta la detection (capitolo 3) e il triage assistito (capitolo 6).

Percorso. Specializzazione Detection & Response (capitoli 3–6).

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- distinguere intelligence strategica, operativa e tattica, e i relativi destinatari;
- estrarre indicatori di compromissione e tecniche da report testuali con strumenti NLP, validandone i risultati;
- giudicare qualità, tempestività e attendibilità di una fonte di intelligence.

4.1 Il ciclo dell'intelligence e i tre livelli

La threat intelligence segue il ciclo classico dell'intelligence. Nella fase di **direzione** si definiscono i requisiti informativi: quali domande l'organizzazione deve porsi sulle minacce che la riguardano. Nella raccolta si acquisiscono dati da feed di indicatori, fonti aperte (OSINT), report di fornitori e agenzie, telemetria interna. Nell'elaborazione i dati vengono normalizzati, tradotti e strutturati, e in questa fase operano le tecniche di NLP. Nell'**analisi** gli analisti valutano i dati, formulano ipotesi e assegnano livelli di confidenza. Nella diffusione i risultati raggiungono i destinatari, in forma di report o di oggetti strutturati scambiati con lo standard STIX attraverso il protocollo TAXII, e il loro riscontro aggiorna i requisiti della fase di direzione.

I prodotti dell'intelligence si distinguono per livello. L'intelligence strategica descrive tendenze e attori per il vertice dell'organizzazione, su orizzonti di mesi. L'intelligence operativa descrive campagne in corso e capacità degli attori, ed è destinata al SOC e al CSIRT su orizzonti di settimane. L'intelligence tattica fornisce TTP e indicatori per la detection: gli indicatori in particolare, soprattutto indirizzi IP e domini, perdono valore in tempi brevi. Alcune classificazioni separano un quarto livello, tecnico, dedicato ai soli indicatori (§ 4.7).

4.2 Elaborazione automatica dei report

Gran parte dell'intelligence disponibile è scritta in prosa: report di fornitori, bollettini di CERT, articoli, post tecnici. Trasformarla in dati strutturati richiede tempo e attenzione, ed è il compito in cui gli strumenti automatici danno il contributo più evidente.

Le espressioni regolari, con verifiche di formato, individuano nel testo gli indicatori: valori di hash, domini, indirizzi IP, identificativi di vulnerabilità (CVE); il riconoscimento di entità (named entity recognition) individua attori, famiglie di malware, strumenti e settori colpiti. La mappatura del testo sulle tecniche ATT&CK associa le descrizioni di comportamento alle tecniche corrispondenti; il progetto open source TRAM (Threat Report ATT&CK Mapper) del Center for Threat-Informed Defense di MITRE ne è un esempio. La sintesi e la deduplicazione riducono la sovrapposizione tra report che descrivono la stessa campagna. L'arricchimento interroga servizi esterni, come VirusTotal o Open Threat Exchange (OTX), per aggiungere informazioni di reputazione e contesto. La ricerca per hash o per indicatore non espone dati dell'organizzazione; il caricamento di file o URL li rende invece consultabili da terzi e può rivelare all'attaccante che il campione è stato individuato.

Tra gli errori tipici, indirizzi e domini citati come esempio vengono scambiati per indicatori, e un modello linguistico può attribuire al testo tecniche che il testo non descrive. Le forme defanged degli indicatori (hxxp al posto di http, [...] al posto del punto), usate per evitare collegamenti attivi, possono non essere riconosciute o essere ricostruite male. Un feed avvelenato con indicatori falsi contamina la detection a valle. Per questo l'estrazione automatica produce **candidati**, e la pubblicazione in piattaforma richiede validazione umana e l'indicazione della confidenza.

4.3 Attendibilità delle fonti e Admiralty Code

L'Admiralty Code, adottato nella dottrina NATO sull'intelligence (AJP-2.1), valuta separatamente la fonte e l'informazione. L'affidabilità della fonte si esprime con una lettera: A (completamente affidabile), B (normalmente affidabile), C (abbastanza affidabile), D (normalmente non affidabile), E (inaffidabile), F (affidabilità non valutabile). La valutazione riguarda la storia della fonte, cioè l'accuratezza delle informazioni fornite in passato, e non il singolo report. La credibilità dell'informazione si esprime con un numero: 1 (confermata da altre fonti), 2 (probabilmente vera), 3 (possibilmente vera), 4 (dubbia), 5 (improbabile), 6 (veridicità non valutabile). Un indicatore valutato B2 è quindi più attendibile di uno valutato D4.

Altri criteri completano la valutazione. La tempestività conta perché un indicatore scaduto genera solo rumore. La rilevanza misura l'attinenza della fonte al proprio settore e al proprio threat model. La completezza riguarda il contesto: una lista di indirizzi senza spiegazioni è molto meno utilizzabile di un indicatore accompagnato dalle TTP e dalla descrizione della campagna. La tracciabilità richiede che la provenienza sia documentata e che la condivisione rispetti le marcature TLP. Un feed importato senza questi filtri porta nel SIEM indicatori scaduti, irrilevanti, privi di contesto o di provenienza ignota.

4.4 Indicatori e TTP nella piramide del dolore

La piramide del dolore, proposta da David Bianco nel 2013 e rivista nel 2014 con l'aggiunta del livello dei valori di hash, ordina gli indicatori secondo il costo che il loro rilevamento impone all'attaccante. Alla base si trovano i valori di hash, che l'attaccante cambia senza sforzo ricompilando o modificando un file, e subito sopra gli indirizzi IP e i nomi di dominio. Seguono gli arte-

fatti di rete e di host, come stringhe caratteristiche nel traffico o chiavi di registro, e gli strumenti. In cima stanno le **TTP** (tattiche, tecniche e procedure), cioè il modo in cui l'avversario opera, che può cambiare solo riprogettando le proprie campagne.

L'investimento in detection rende di più ai livelli alti, dove il valore persiste nel tempo. Il blocco dell'indirizzo IP del server di comando e controllo in uso oggi interrompe una campagna per poco tempo. Una detection sulla tecnica, per esempio l'esecuzione di DLL da percorsi utente tramite rundll32, intercetta anche le varianti successive che continuano a usarla. La piramide spiega anche la scelta, nel capitolo 3, di misurare la copertura in tecniche ATT&CK.

Per l'estrazione automatica, gli indicatori si ricavano con espressioni regolari e verifiche di formato, mentre le TTP sono descritte nella prosa dei report; su questa parte l'estrazione assistita offre il contributo maggiore e commette più errori.

4.5 STIX, TAXII e TLP per la condivisione

STIX (Structured Threat Information Expression), nella versione 2.1 standardizzata da OASIS nel 2021, è il formato per rappresentare l'intelligence. Usa oggetti JSON tipizzati, come indicator, malware, threat-actor, attack-pattern e campaign, collegati da relazioni esplicite: un indicatore *indica* un malware, che è *usato da* un attore. **TAXII** (Trusted Automated Exchange of Intelligence Information), versione 2.1, è il protocollo con cui un client invia oggetti STIX a una raccolta (collection) di un server o li richiede, tramite API HTTPS; i canali di tipo publish-subscribe sono previsti dalla specifica ma non ancora definiti.

Il Traffic Light Protocol (TLP), mantenuto dal FIRST e giunto alla versione 2.0 nel 2022, indica i limiti di ridistribuzione di un'informazione, che chi la riceve è tenuto a rispettare. I livelli vanno da TLP:RED, che riserva l'informazione ai destinatari diretti, a TLP:CLEAR, che non pone limiti alla diffusione. Tra i due estremi, TLP:GREEN ammette la condivisione nella comunità di riferimento e TLP:AMBER solo con chi ha necessità di conoscerla, all'interno della propria organizzazione e con i suoi clienti, mentre TLP:AMBER+STRICT la limita alla sola organizzazione.

Scenario didattico. Un CSIRT nazionale pubblica in una raccolta TAXII un bundle STIX marcato TLP:AMBER su una campagna contro il settore sanitario. Le strutture autorizzate lo scaricano e lo importano in MISP, e le relazioni tra attore, tecniche e indicatori arrivano già strutturate per la detection.

Le relazioni trasformano un elenco di indicatori in conoscenza interrogabile: una richiesta come «tutte le tecniche osservate per l'attore X negli ultimi sei mesi» è una query sul grafo degli oggetti, impossibile su un feed di soli indicatori.

4.6 Errori tipici dell'estrazione automatica

Scenario didattico. Un report scrive: «l'attaccante ha usato un dominio simile a update-vendor[.]example; per il test interno abbiamo riprodotto il traffico verso 192.168.10.5». Un estrattore che non analizza il contesto ne ricava due indicatori errati: il dominio ricostruito dalla forma defanged, che potrebbe essere legittimo o già neutralizzato con un sinkhole, e un indirizzo privato, appartenente agli intervalli riservati dalla RFC 1918, usato nell'ambiente di test dell'autore. Una volta pubblicati in piattaforma, il dominio genera blocchi errati e l'indirizzo privato produce falsi riscontri interni nel SIEM.

Prima della pubblicazione occorre quindi verificare il contesto della frase (attacco osservato o esempio), ricostruire e controllare il defanging, escludere gli intervalli di indirizzi riservati e i domini di infrastrutture comuni, considerare la data del report e assegnare confidenza e marcatura TLP.

Sul piano organizzativo, l'output dell'estrazione automatica entra in piattaforma come bozza non pubblicata e la promozione a evento condivisibile è un atto umano registrato. Un indicatore falso condiviso compromette la credibilità della fonte e la detection di tutti i destinatari.

4.7 Livelli dell'intelligence e destinatari

Il rapporto pubblicato nel 2015 da MWR InfoSecurity, con la collaborazione del CPNI e del CERT-UK britannici, propone una classificazione in quattro livelli, che separa il livello tattico da quello tecnico. Il livello **strategico** (tendenze, attori, contesto geopolitico) serve al vertice per le decisioni di investimento e di rischio. Il livello operativo orienta le priorità del SOC e del CSIRT. Nel rapporto MWR comprende informazioni su attacchi specifici e imminenti contro l'organizzazione, destinate ai responsabili della sicurezza e della risposta agli incidenti e descritte come molto rare; in classificazioni successive designa le campagne in corso e le capacità degli attori rilevanti per il settore. Il livello tattico (TTP mappate su ATT&CK) è la base del detection engineering. Il livello tecnico (indicatori come hash, indirizzi IP e domini) alimenta direttamente i controlli ed è il più deperibile.

Scenario didattico. La stessa campagna ransomware produce una nota di una pagina per il vertice, con il settore colpito, l'impatto e lo stato delle difese, e un briefing operativo per il SOC, con i vettori iniziali osservati e i tempi tipici tra accesso e cifratura. Ne derivano anche una mappa delle TTP per i detection engineer e un pacchetto STIX di indicatori per il blocco automatico.

La trasposizione tra livelli si presta all'uso di modelli linguistici, per esempio per passare dal report tecnico alla nota per il vertice. Il rischio è che la sintesi perda le espressioni di confidenza dell'analista, come «valutiamo con bassa confidenza che», e presenti come certe conclusioni probabilistiche, un errore che si individua rileggendo la sintesi accanto al testo originale.

4.8 Attribuzione e livelli di confidenza

L'attribuzione procede per gradi di difficoltà crescente. Raggruppare attività in **cluster** coerenti per strumenti, infrastrutture e orari è relativamente solido, mentre collegare un cluster a un attore nominato è già più incerto. Per arrivare a un'organizzazione o a uno Stato occorre combinare evidenze tecniche, contesto e spesso informazioni non disponibili ai privati.

Ogni indizio può essere falsificato, perché codice e infrastrutture si riusano o si condividono, gli orari di attività si possono simulare e l'inserimento deliberato di indizi fuorvianti, detto **false flag**, è documentato. L'analisi usa quindi un linguaggio di confidenza calibrato (bassa, media, alta) e separa l'osservazione, per esempio «il loader condivide il 60% del codice con il malware X», dall'inferenza, per esempio «riteniamo con confidenza media che l'attività sia riconducibile allo stesso gruppo».

Caso documentato. Olympic Destroyer, il malware usato nel 2018 contro l'organizzazione dei Giochi olimpici invernali di PyeongChang, conteneva artefatti che rimandavano a più attori statali, inseriti come false flag. Cisco Talos ne descrisse il funzionamento e Kaspersky dimostrò che uno degli artefatti, l'intestazione Rich di un eseguibile, era stato falsificato per rimandare al gruppo Lazarus; nel 2020 il Dipartimento di Giustizia degli Stati Uniti ha incriminato ufficiali dell'intelligence militare russa (GRU) per questa e altre operazioni.

Per la risposta aziendale serve soprattutto il profilo operativo dell'avversario: se si tratti di un ransomware opportunistico o di un attore mirato, quali azioni sia probabile che compia in seguito. L'attribuzione a uno Stato ha implicazioni legali e politiche: nelle comunicazioni ufficiali l'organizzazione riporta i fatti tecnici e lascia l'attribuzione alle autorità competenti. Lo stesso metodo, con l'osservazione separata dall'inferenza e la confidenza dichiarata, si applica alle relazioni forensi (capitoli 7 e 8).

4.9 Condivisione dell'intelligence tra organizzazioni

Nessuna organizzazione osserva abbastanza attacchi da costruire da sola un quadro completo della minaccia, e la condivisione ne amplia la capacità di difesa. Le strutture principali sono gli **ISAC** (Information Sharing and Analysis Center) settoriali e i CSIRT nazionali, in Italia il CSIRT Italia presso l'Agenzia per la cybersicurezza nazionale, che operano come comunità di fiducia. MISP, piattaforma open source sviluppata con il contributo del CERT lussemburghese CIRCL, è tra gli strumenti più diffusi: organizza le informazioni in eventi, attributi e tag, e sincronizza le istanze di organizzazioni diverse.

Condividere ha un costo per chi lo fa: tempo di cura dei dati, rischio reputazionale, timore di rivelare una compromissione. Il beneficio va soprattutto agli altri membri della comunità. Le comunità efficaci riducono il costo con modelli predefiniti, anonimizzazione e automazione, e chiedono uno scambio nei due sensi.

Scenario didattico. Una banca subisce un tentativo di frode con un kit nuovo e condivide sull'ISAC di settore, in TLP:AMBER e in forma anonimizzata, indicatori e TTP entro 24 ore. Altri tre istituti trovano gli stessi indicatori nei propri log prima di subire danni.

Un feed si valuta per tempestività, tasso di falsi positivi sul traffico della propria organizzazione, contesto allegato a ogni indicatore e sovrapposizione con le fonti già disponibili. L'AI aiuta a deduplicare e arricchire i dati; un numero maggiore di indicatori comporta anche più lavoro di verifica.

4.10 Laboratorio: pipeline di intelligence in MISP o OpenCTI

La pipeline si costruisce su una piattaforma di threat intelligence open source, MISP oppure OpenCTI. Un report pubblico su un attore APT viene importato nella piattaforma e un modello linguistico eseguito in locale con Ollama, senza che i dati escano dalla macchina, ne estrae indicatori e tecniche. I risultati si validano a mano, verificando contesto, defanging e falsi indicatori secondo la procedura del § 4.6. Si pubblica infine un evento MISP (o un report OpenCTI) esportabile come bundle STIX 2.1, con la marcatura TLP corretta. STIX 2.1 prevede nativamente le marcature TLP 1.0, mentre quelle del TLP 2.0 si rappresentano con un'estensione dedicata.

Consegna: evento o report pubblicato con indicatori validati; tabella degli errori dell'estrazione automatica, classificati per tipo; mappatura ATT&CK delle tecniche descritte nel report. Dal confronto in plenaria dei tassi di errore ottenuti dai gruppi emerge l'effetto delle istruzioni date al modello.

4.11 Letture

- B. Strom et al., *MITRE ATT&CK: Design and Philosophy*, MITRE, 2018 (rev. 2020).
- OASIS, *STIX Version 2.1* e *TAXII Version 2.1*, OASIS Standard, 2021.
- FIRST, *Traffic Light Protocol (TLP) Version 2.0*, 2022.
- D. Bianco, «The Pyramid of Pain», 2013 (rev. 2014).
- D. Chismon, M. Ruks, *Threat Intelligence: Collecting, Analysing, Evaluating*, MWR InfoSecurity, con il contributo di CPNI e CERT-UK, 2015.
- NATO, *AJP-2.1 Allied Joint Doctrine for Intelligence Procedures*, edizione corrente.
- Cisco Talos, «Olympic Destroyer Takes Aim At Winter Olympics», 12 febbraio 2018.
- Kaspersky GReAT, «The devil's in the Rich header», 8 marzo 2018.
- U.S. Department of Justice, «Six Russian GRU Officers Charged in Connection with Worldwide Deployment of Destructive Malware and Other Disruptive Actions in Cyberspace», 19 ottobre 2020.
- Un report pubblico recente a scelta (Mandiant, ESET o un advisory della CISA).

4.12 Domande di autovalutazione

1. Quali rischi comporta l'ingestione automatica di feed non verificati? *Traccia.* Indicatori scaduti, errati o deliberatamente falsi entrano nella detection e producono falsi positivi, blocchi di risorse legittime e perdita di fiducia nel sistema; un feed avvelenato può anche essere usato per indurre l'organizzazione a bloccare servizi propri.
2. Che differenza c'è tra IoC e TTP in termini di persistenza nel tempo? *Traccia.* Gli indicatori (hash, indirizzi, domini) si cambiano con poco sforzo e perdono valore rapidamente; le TTP descrivono il modo di operare dell'attaccante e cambiano solo con la riprogettazione delle campagne, come illustra la piramide del dolore.
3. Come si valuta l'attendibilità di una fonte con l'Admiralty Code? *Traccia.* Si assegnano due valori indipendenti: una lettera da A a F per l'affidabilità della fonte, basata sulla sua storia, e un numero da 1 a 6 per la credibilità della singola informazione, basata sulle conferme disponibili.
4. Quali errori tipici commette un LLM nell'estrazione di IoC? *Traccia.* Scambia per indicatori gli esempi e gli indirizzi dell'ambiente di test, non riconosce o ricostruisce male le forme defanged, attribuisce al testo tecniche che non vi sono descritte e omette le espressioni di confidenza dell'autore.
5. A cosa servono STIX e TAXII? *Traccia.* STIX rappresenta l'intelligence come oggetti tipizzati collegati da relazioni, interrogabili come un grafo; TAXII è il protocollo con cui un client invia oggetti STIX a una raccolta di un server o li richiede, e automatizza lo scambio tra organizzazioni.

Capitolo 5. SOAR e automazione della risposta

Il capitolo tratta l'orchestrazione e l'automazione della risposta agli incidenti. Dopo aver descritto i livelli di autonomia che un sistema può avere rispetto all'operatore umano, analizza la struttura di un playbook automatizzato e i rischi che l'automazione introduce. Gli approfondimenti riguardano i criteri per scegliere il livello di autonomia di ogni azione, i limiti del controllo umano (automation bias), la progettazione dell'audit trail e del punto di approvazione, le metriche con cui valutare l'automazione e l'uso dei playbook per acquisire le evidenze nelle prime fasi della risposta. Il capitolo collega la detection e la threat intelligence trattate nei capitoli 3 e 4 alla risposta, e prepara il capitolo 6, in cui alcune decisioni vengono affidate a modelli linguistici.

Percorso. Specializzazione Detection & Response (capitoli 3–6).

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- classificare i livelli di autonomia di un sistema di risposta (human-in-the-loop, human-on-the-loop, human-out-of-the-loop) e attribuirli alle singole azioni;
- progettare un playbook SOAR con punti di approvazione umana, gestione degli errori e rollback;
- valutare costi e rischi dell'automazione di azioni irreversibili o ad alto impatto.

5.1 Tre livelli di autonomia nella risposta

Con l'acronimo **SOAR** (Security Orchestration, Automation and Response) si indicano le piattaforme che coordinano gli strumenti di sicurezza e ne automatizzano l'uso. L'orchestrazione integra tramite API sistemi diversi (SIEM, EDR, firewall, gateway di posta, piattaforme di threat intelligence, sistemi di ticketing) in un unico flusso di lavoro. Con l'automazione i passi di un runbook vengono eseguiti senza intervento manuale, e la gestione dei casi raccoglie in un fascicolo unico alert, osservabili, azioni e comunicazioni. In laboratorio queste funzioni si studiano con strumenti gratuiti come Shuffle (flussi di lavoro, open source), TheHive (gestione dei casi, distribuito da StrangeBee con una licenza Community gratuita e codice non pubblico) e Cortex (analisi automatica degli osservabili, open source).

Il tema centrale dell'automazione è la posizione dell'essere umano rispetto alla decisione. Si distinguono tre livelli:

- **human-in-the-loop:** il sistema propone, l'operatore approva esplicitamente ogni azione prima dell'esecuzione;
- **human-on-the-loop:** il sistema agisce, l'operatore osserva in tempo reale e può interrompere o annullare;
- **human-out-of-the-loop:** il sistema agisce senza intervento umano.

Il livello si sceglie per ciascuna azione e non per l'intera piattaforma. Lo stesso playbook può arricchire gli indicatori in piena autonomia, mettere in quarantena un messaggio sotto supervisione e attendere un'approvazione per bloccare un account. La scelta dipende dalla reversibilità dell'azione (§ 5.4), dal suo impatto e dall'urgenza, perché un'approvazione richiede tempo e in alcune situazioni il ritardo causa a sua volta un danno.

La nozione di controllo umano significativo, nata nel dibattito sui sistemi autonomi, è oggi ripresa dalla regolazione dell'intelligenza artificiale: l'art. 14 del Regolamento (UE) 2024/1689 (AI Act) richiede che i sistemi di AI ad alto rischio siano progettati per consentire una sorveglianza umana efficace. Per i sistemi ad alto rischio dell'allegato III questi obblighi si applicano dal 2 dicembre 2027, data fissata dal Regolamento (UE) 2026/1744. Un playbook composto da regole definite da persone non è di per sé un sistema di AI (considerando 12), e i componenti destinati unicamente alla cibersecurity non sono considerati componenti di sicurezza delle infrastrutture critiche ai fini dell'allegato III, punto 2 (considerando 55); il principio resta comunque un criterio di progetto. Il capitolo 15 riprende il tema dal punto di vista giuridico.

5.2 Anatomia di un playbook di risposta al phishing

Un playbook automatizzato descrive una sequenza di passi con condizioni e punti di decisione. Il caso del phishing segnalato da un utente è un esempio tipico:

- **trigger:** la segnalazione dell'utente o un alert del gateway di posta avvia il flusso;
- **analisi:** gli allegati vengono eseguiti in una sandbox e gli URL verificati con servizi di reputazione;
- **arricchimento:** gli indicatori estratti (domini, indirizzi IP, hash) sono confrontati con le fonti di threat intelligence e si cercano altri destinatari dello stesso messaggio;
- **approvazione:** l'operatore decide sul blocco del mittente e sulla quarantena dei messaggi;
- **azione:** quarantena, blocco, apertura del ticket, notifica agli utenti coinvolti, con registrazione di ogni passo.

Dal punto di vista ingegneristico un playbook è software e ne condivide i requisiti. Ogni passo deve gestire gli errori (un servizio esterno che non risponde, un risultato incompleto), e le azioni devono essere idempotenti, cioè produrre lo stesso effetto se ripetute. Il rollback va definito e provato prima della messa in esercizio, e ogni decisione, automatica o umana, viene registrata con orario e motivazione.

I playbook sono in gran parte espressi nei formati proprietari delle singole piattaforme. La specifica OASIS CACAO (Collaborative Automated Course of Action Operations) Security Playbooks, nella versione 2.0 approvata come Committee Specification nel 2023, definisce un formato JSON per descrivere i playbook indipendentemente dal prodotto, così da poterli condividere tra organizzazioni e sottoporre a verifiche sistematiche.

5.3 Rischi dell'automazione della risposta

L'automazione riduce i tempi di risposta e ripete un errore di logica o di classificazione a ogni esecuzione. I principali modi di fallimento sono:

- **contenimento su falso positivo:** un'azione automatica blocca un servizio legittimo, con un danno che può superare quello dell'incidente ipotizzato;
- **reazioni a catena:** un playbook ne innesca un altro, che a sua volta ne innesca un terzo, fino a cicli che saturano sistemi e analisti;
- **automation bias:** l'operatore approva senza verificare (§ 5.5);
- **abuso da parte dell'attaccante:** chi conosce le regole di risposta può provocare deliberatamente blocchi automatici, per esempio facendo apparire come malevolo un dominio legittimo, e ottenere un disservizio senza compromettere alcun sistema.

Le contromisure sono di progetto. Le azioni irreversibili o ad alto impatto richiedono un'approvazione umana, e il rollback viene progettato e provato prima del rilascio. Meccanismi di limitazione della frequenza (rate limiting) e interruttori automatici (circuit breaker) fermano un playbook che esegue troppe azioni in poco tempo. Gli elenchi di risorse critiche escludono dalle azioni automatiche i sistemi il cui blocco avrebbe conseguenze gravi. Infine le metriche devono contare gli errori causati dall'automazione insieme al tempo risparmiato e agli errori evitati (§ 5.7). L'uso dell'automazione difensiva come superficie d'attacco è ripreso nel capitolo 13.

5.4 Reversibilità delle azioni e livello di autonomia

Un'azione di risposta è **reversibile** quando esiste una procedura nota, verificata e rapida per annullarne gli effetti senza perdite. La reversibilità dipende dal contesto oltre che dall'azione: isolare dalla rete una workstation d'ufficio è un'operazione reversibile, mentre isolare la stessa macchina quando controlla una linea di produzione, come avviene negli ambienti industriali (OT), può interrompere un processo fisico.

Per ogni azione del playbook occorre stabilire come si annulla, in quanto tempo e quali effetti produce nel frattempo. Ne deriva una scala pratica:

- arricchire un indicatore con interrogazioni passive su fonti approvate non ha effetti sui sistemi e può avvenire in autonomia, mentre l'invio di file o URL a servizi esterni, che può divulgare dati riservati o rivelare all'attaccante di essere stato individuato, segue la policy sui dati che escono dall'organizzazione (§ 6.6);
- mettere in quarantena un messaggio si annulla con una singola operazione: autonomia con notifica;
- bloccare un account è reversibile ma interrompe il lavoro di una persona: richiede approvazione;
- revocare certificati o cancellare da remoto un dispositivo sono operazioni di fatto irreversibili: richiedono una decisione umana.

Un metodo utile è il test del falso positivo: prima di automatizzare un'azione si simula il suo innesco su un falso positivo nel momento più sfavorevole, per esempio sull'account del direttore finanziario durante la chiusura di bilancio. Se il danno risultante è inaccettabile, l'azione resta subordinata all'approvazione.

5.5 Automation bias nei punti di approvazione

L'**automation bias** è la tendenza ad affidarsi all'indicazione di un sistema automatico anche quando altre informazioni disponibili la contraddicono. È stato studiato a lungo in aviazione, in medicina e nel controllo di processo; la letteratura distingue gli errori di omissione (non accorgersi di un problema che il sistema non segnala) dagli errori di commissione (seguire un'indicazione errata del sistema).

Nel SOC il fenomeno si manifesta nei punti di approvazione, insieme alla **complacency**, cioè il calo della vigilanza dovuto alla fiducia nel sistema, che Parasuraman e Manzey descrivono come fenomeno distinto ma sovrapposto. Dopo molte proposte corrette, l'operatore tende ad approvare le successive senza esaminarle, e il controllo umano previsto dal processo si riduce a una formalità.

Scenario didattico. Il playbook propone il blocco di un mittente classificato come malevolo; l'analista, al quarantesimo caso della giornata, approva in pochi secondi. Il dominio apparteneva a un fornitore segnalato per errore e il flusso degli ordini si interrompe per alcune ore.

Il problema si affronta nel disegno del processo e dell'interfaccia, oltre che con la formazione. Tra le contromisure figurano la presentazione all'approvatore delle evidenze insieme alla raccomandazione, la motivazione scritta per le approvazioni ad alto impatto e i controlli a campione con casi di prova noti; il tempo medio di approvazione ne misura l'effetto, perché tempi molto brevi e costanti indicano un controllo inefficace.

5.6 Progettazione dell'audit trail

L'audit trail è il registro delle azioni eseguite dal sistema e dagli operatori. Con la sola annotazione «azione eseguita» una decisione non si ricostruisce né si giustifica. Per ogni azione il trail registra che cosa è stato fatto e quando, con orologi sincronizzati tra i sistemi, e su quale base, cioè l>alert, le evidenze e la versione del playbook in uso. Registra inoltre chi ha deciso, il sistema in autonomia oppure un operatore identificato, e se l'azione è riuscita, è fallita o è stata annullata con un rollback.

Il trail va progettato insieme al playbook e verificato durante i test. Dal punto di vista tecnico deve essere a sola aggiunta (append-only) e protetto da modifiche anche da parte degli amministratori della piattaforma. Il suo valore probatorio dipende dalla coerenza dei timestamp tra i sistemi, e i tempi di conservazione seguono gli obblighi applicabili e la policy interna.

Scenario didattico. Sei mesi dopo un blocco di account contestato da un dipendente, il trail mostra l>alert con il relativo punteggio, le evidenze allegate, l'approvazione dell'analista con la motivazione e il rollback eseguito in giornata. La decisione si ricostruisce e si può motivare davanti all'organizzazione e, se necessario, in giudizio. Lo stesso schema di registrazione si applica ai log degli agenti basati su modelli linguistici (capitolo 14).

5.7 Metriche per valutare l'automazione

Il numero di playbook automatizzati è un dato di inventario. Le misure utili per valutare l'automazione sono:

- il **MTTR per classe di incidente**, confrontato prima e dopo l'introduzione dell'automazione su casi dello stesso tipo;
- il tempo di lavoro degli analisti risparmiato per ogni esecuzione, misurato e non stimato;
- il **tasso di override**, cioè la frequenza con cui l'operatore corregge la proposta dell'automazione;
- gli errori causati dall'automazione, contati con lo stesso rigore degli errori evitati.

Un tasso di override vicino a zero va verificato: può indicare un sistema accurato oppure un controllo umano non più esercitato (§ 5.5).

Scenario didattico. Si supponga che, dopo sei mesi, il playbook anti-phishing di un SOC mostri un MTTR passato da circa tre ore a poco più di venti minuti, un tasso di override del 6% concentrato sulla decisione di blocco del mittente e due errori causati, entrambi annullati entro un'ora. Su questi dati il gruppo può discutere se aumentare l'autonomia dell'azione di blocco.

Una metrica di velocità usata da sola favorisce chiusure affrettate. Va affiancata da indicatori di qualità, come il tasso di riapertura dei casi e il numero di escalation mancate, secondo il principio già visto per le metriche di risposta (capitolo 1).

5.8 Progettazione del punto di approvazione

Un'approvazione è efficace se chi decide dispone del contesto: che cosa è accaduto, che cosa farà l'azione, quali persone e sistemi coinvolge, quale rischio comporta non agire, con le evidenze consultabili senza cambiare strumento. Il progetto deve inoltre definire:

- il **timeout**, cioè che cosa accade se nessuno risponde entro il termine: escalation a un altro ruolo, esecuzione di un'azione conservativa predefinita o nessuna azione, scelte che comportano rischi diversi;
- i ruoli autorizzati ad approvare in funzione dell'impatto, così che un analista di primo livello non autorizzi l'isolamento di un sistema critico;
- la registrazione della decisione, della motivazione e dell'orario.

Scenario didattico. La richiesta di blocco di un account arriva al canale di reperibilità come scheda con l'utente, i segnali che l'hanno generata, i sistemi raggiungibili dall'account, il rollback previsto e due pulsanti con le conseguenze descritte. Dopo dieci minuti senza risposta la richiesta passa al turno senior; se nessuno risponde, si revocano soltanto le sessioni attive dell'account.

Gli errori frequenti sono le approvazioni in serie senza contesto, la presenza di un unico approvatore possibile, che diventa collo di bottiglia e punto singolo di guasto, e il silenzio interpretato come assenso. Il disegno dell'interfaccia di approvazione è una delle contromisure all'automation bias (§ 5.5).

5.9 Acquisizione automatica delle evidenze

L'automazione serve anche a preservare le evidenze. Le evidenze più volatili, come il contenuto della memoria e lo stato delle connessioni (capitolo 7), si perdono nei minuti in cui il gruppo di risposta si sta ancora organizzando. Un playbook con acquisizione iniziale integrata raccoglie all'innesco, di norma prima del contenimento:

- l'immagine della memoria dell'host o del processo sospetto;
- un pacchetto di triage (processi, connessioni, voci di avvio automatico, log principali);
- l'esportazione dei log cloud rilevanti.

Ogni elemento viene accompagnato da hash e timestamp e depositato in un archivio non modificabile collegato al caso, e l'accaduto resta ricostruibile anche dopo che il contenimento ha alterato lo stato dei sistemi.

Scenario didattico. Alle 04:12 un alert EDR segnala un server. Il playbook acquisisce l'immagine del processo, il pacchetto di triage e l'elenco delle connessioni, calcola gli hash e deposita i file nell'archivio del caso; poi isola l'host. Quando l'analista si collega, il contenimento è avvenuto e le evidenze di quel momento sono disponibili.

L'acquisizione automatica è a sua volta un'azione sul sistema. Strumenti, versioni e orari vanno registrati nel log del playbook, perché in sede giudiziaria occorrerà spiegare come le evidenze sono state raccolte (capitolo 7); l'archivio va protetto quanto i dati che contiene. L'ordine tra acquisizione e contenimento si stabilisce per classe di incidente: con un'attività distruttiva in corso, per esempio la cifratura dei dati da parte di un ransomware, il contenimento può precedere l'acquisizione della memoria. L'isolamento di rete tramite EDR mantiene la connessione con la console, e la raccolta si completa dopo l'isolamento; l'elenco delle connessioni attive si acquisisce prima. Il tema riprende l'ordine di volatilità del capitolo 7 (§ 7.6) e prosegue, nella specializzazione Digital Forensics, nel capitolo 9.

5.10 Laboratorio: playbook anti-phishing con Shuffle e TheHive

Il flusso descritto nel § 5.2 si realizza con strumenti gratuiti: Shuffle (open source) per l'orchestrazione, TheHive (licenza Community) per la gestione dei casi e Cortex (open source) per l'analisi automatica degli osservabili.

I gruppi procedono in quest'ordine:

1. costruzione del flusso: trigger, analisi con gli analyzer di Cortex, arricchimento degli indicatori;

2. inserimento del punto di approvazione umana per il blocco del mittente, progettato secondo il § 5.8;
3. apertura automatica del caso in TheHive con gli osservabili estratti;
4. prova su messaggi simulati, benigni e malevoli, e verifica dell'audit trail.

Consegna: il playbook funzionante ed esportato; la matrice motivata tra azioni e livelli di autonomia; la prova del rollback su un'azione di blocco. Nella valutazione la matrice pesa più del flusso, perché documenta il ragionamento del gruppo.

5.11 Letture

- NIST, *SP 800-61 Rev. 3, Incident Response Recommendations and Considerations for Cybersecurity Risk Management: A CSF 2.0 Community Profile*, 2025.
- OASIS, *CACAO Security Playbooks Version 2.0*, Committee Specification 01, 2023.
- R. Parasuraman, D. H. Manzey, «Complacency and Bias in Human Use of Automation: An Attentional Integration», *Human Factors*, 52(3), 2010.
- R. Parasuraman, V. Riley, «Humans and Automation: Use, Misuse, Disuse, Abuse», *Human Factors*, 39(2), 1997.
- Documentazione ufficiale di TheHive e Cortex (StrangeBee) e di Shuffle.
- Regolamento (UE) 2024/1689 (AI Act), art. 14, nel testo modificato dal Regolamento (UE) 2026/1744.

5.12 Domande di autovalutazione

1. Quali azioni di risposta non dovrebbero mai essere pienamente automatizzate, e perché? *Traccia.* Le azioni irreversibili o ad alto impatto, come la cancellazione remota di dispositivi, la revoca di certificati o l'isolamento di sistemi che controllano processi fisici. Un falso positivo produrrebbe danni non annullabili; la decisione richiede una valutazione umana del contesto.
2. Cosa distingue human-in-the-loop da human-on-the-loop? *Traccia.* Nel primo caso l'azione attende un'approvazione esplicita prima dell'esecuzione; nel secondo il sistema agisce e l'operatore supervisiona, con la possibilità di interrompere o annullare. Il secondo è adatto ad azioni reversibili e urgenti.
3. Come si progetta il rollback di un contenimento errato? *Traccia.* Per ogni azione si definisce prima del rilascio la procedura inversa, la si automatizza dove possibile, la si prova su casi simulati e se ne registra l'esecuzione nell'audit trail. Occorre conoscere anche gli effetti prodotti nel tempo in cui l'azione è rimasta attiva.
4. Quali metriche giustificano l'investimento in SOAR? *Traccia.* MTTR per classe di incidente prima e dopo l'automazione, tempo di lavoro risparmiato misurato, tasso di override, errori causati ed evitati, affiancati da indicatori di qualità come le riaperture dei casi.
5. In quali casi l'audit trail è anche un requisito giuridico? *Traccia.* L'audit trail documenta chi ha deciso, su quali basi e con quale esito, ed è quindi necessario per attribuire responsabilità, rispondere a contestazioni e dimostrare il controllo umano dove le norme lo

richiedono. L'art. 22 GDPR riguarda le decisioni basate unicamente su un trattamento automatizzato che producono effetti giuridici o incidono in modo analogo significativamente sulla persona, ipotesi che può riguardare, per esempio, il blocco dell'account di un dipendente. Gli artt. 12, 14 e 26 AI Act valgono per i soli sistemi di AI ad alto rischio, categoria in cui di regola non rientrano gli strumenti destinati unicamente alla cibersicurezza (considerando 55); per i sistemi dell'allegato III i loro obblighi si applicano dal 2 dicembre 2027 (Regolamento (UE) 2026/1744). Il tema è sviluppato nel capitolo 15.

Capitolo 6. LLM e agenti nel SOC

Il capitolo esamina l'impiego dei modelli linguistici di grandi dimensioni (LLM) nelle attività di un Security Operations Center. Considera le forme di impiego in ordine di autonomia crescente: il copilota che assiste l'analista, il modello che genera query e regole di detection, l'agente che conduce in autonomia una parte del triage. Per ciascuna descrive le capacità oggi consolidate, i limiti strutturali e le condizioni d'uso. Gli approfondimenti riguardano la struttura del prompt, le allucinazioni e la loro misura, la scelta tra modelli locali e servizi cloud, le alternative tra prompt, retrieval e fine-tuning, la progettazione dei permessi degli agenti e un criterio generale per scegliere i compiti da affidare ai modelli. Il capitolo applica il controllo umano discusso nel capitolo 5 e anticipa i temi di sicurezza dei sistemi di AI trattati nei capitoli 13 e 14.

Percorso. Specializzazioni Detection & Response (capitoli 3–6) e Sicurezza dell'AI (capitoli 6 e 12–14). Per chi segue Sicurezza dell'AI, letture di raccordo: § 3.6, 3.9 e 5.1.

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- usare un copilota per accelerare un'indagine mantenendo la verifica umana delle sue affermazioni;
- generare con un LLM query KQL o SPL e regole Sigma, e validarle prima della messa in produzione;
- valutare benefici e rischi dell'autonomia agentica nel triage di primo livello.

6.1 Capacità e limiti dei copiloti investigativi

Un copilota investigativo è un sistema che combina un modello linguistico con connettori verso i dati dell'organizzazione: SIEM, piattaforme EDR, sistemi di gestione delle identità, fonti di threat intelligence. Quando l'analista pone una domanda, il sistema recupera i dati pertinenti attraverso le API dei connettori e li inserisce nel contesto del modello, che formula la risposta. Questo ancoraggio della risposta ai dati dell'organizzazione è detto **grounding**. Prodotti come Microsoft Security Copilot e le funzioni basate su Gemini in Google Security Operations seguono questa architettura; i nomi e le funzioni cambiano rapidamente, per cui il capitolo li cita solo come esempi.

Le capacità oggi consolidate riguardano:

- la sintesi di un incidente a partire da alert e log eterogenei;
- la spiegazione di script e comandi sospetti, compresi quelli offuscati;
- la traduzione di richieste in linguaggio naturale in query, per esempio KQL per gli ambienti Microsoft o SPL per Splunk;
- la redazione di bozze di report e di comunicazioni.

I limiti sono strutturali. Il modello può produrre **allucinazioni**, cioè affermazioni plausibili ma prive di fondamento nei dati (§ 6.5). La groundedness, ossia la misura in cui le affermazioni sono riconducibili alle evidenze fornite, va verificata caso per caso. L'invio di dati riservati a un

servizio cloud di terze parti solleva questioni contrattuali e di protezione dei dati personali (§ 6.6). Per i casi d'uso sensibili un'alternativa è l'uso di modelli a pesi aperti (open weight) eseguiti in locale con strumenti come Ollama o vLLM, con capacità in genere inferiori e pieno controllo sui dati.

6.2 Regole di detection generate da un LLM

Il detection engineering assistito da un LLM procede per passi successivi.

1. *Generazione.* A partire dalla descrizione di una tecnica d'attacco, per esempio una tecnica del framework MITRE ATT&CK, il modello produce una regola in formato Sigma, il formato aperto per descrivere regole di detection su log indipendentemente dal SIEM.
2. *Conversione.* Lo strumento sigma-cli, basato sulla libreria pySigma, traduce la regola nel linguaggio di interrogazione del SIEM di destinazione.
3. *Validazione.* La regola viene provata su log noti: log prodotti da un attacco simulato, che la regola deve rilevare, e traffico benigno rappresentativo, sul quale si misurano i falsi positivi.
4. *Rilascio.* La regola entra in un repository versionato, passa una revisione tra pari e viene monitorata in produzione, in particolare per il tasso di falsi positivi.

Le regole generate presentano errori ricorrenti. Alcune citano campi inesistenti nello schema dei log in uso, perché i nomi dei campi cambiano tra Sysmon, schemi normalizzati e prodotti diversi. Altre hanno condizioni troppo larghe, che producono molti falsi positivi, o logiche troppo strette, adattate al solo esempio fornito. In altri casi la sintassi è valida ma la semantica è errata, per esempio quando la regola interroga una sorgente di log sbagliata. Per questo la validazione del terzo passo è obbligatoria. Il ciclo completo di detection engineering in cui queste regole si inseriscono è descritto nel § 3.9.

6.3 Agenti nel triage di livello 1

Un agente è un modello linguistico che pianifica un'attività in più passi e li esegue chiamando strumenti: interroga il SIEM, arricchisce gli indicatori, consulta l'inventario degli asset, formula una conclusione (§ 6.8). Il triage di primo livello contiene compiti adatti a questa forma di automazione: la deduplicazione degli alert, la chiusura motivata di falsi positivi ricorrenti, la raccolta del contesto prima che un analista prenda in carico il caso. L'obiettivo è spostare il lavoro degli analisti verso la ricerca proattiva delle minacce (threat hunting) e i casi complessi.

L'impiego di agenti richiede condizioni precise:

- strumenti in sola lettura per impostazione predefinita, con le azioni che modificano lo stato dei sistemi subordinate ad approvazione (capitolo 5);
- privilegio minimo per ogni strumento e account di servizio dedicati;
- registrazione completa di piani, chiamate e motivazioni, necessaria anche per l'indagine sugli errori dell'agente (capitolo 14);
- misura della groundedness e controllo umano a campione degli esiti.

La lettura di contenuti non fidati, come i messaggi di posta da analizzare, espone l'agente alla prompt injection indiretta, trattata nel capitolo 13, che trasforma gli strumenti dell'agente in una superficie d'attacco.

6.4 Struttura del prompt operativo

Un copilota SOC si interroga con un prompt strutturato in tre parti. Il **grounding** fornisce le evidenze su cui il modello deve basarsi: log, alert, estratti di regole, inseriti direttamente nel prompt o recuperati con tecniche di retrieval (capitolo 14). Il contesto di ruolo descrive il compito richiesto: triage, formulazione di ipotesi, scrittura di query di verifica. Il **formato vincolato** stabilisce i campi dell'output, la citazione delle righe di log usate e il livello di confidenza.

Se manca il grounding, il modello completa la risposta con conoscenze generali apprese in addestramento, e un output privo di formato vincolato non si può verificare né passare ad altri strumenti.

Esempio. «Sei un analista di secondo livello. Dati questi 40 eventi Sysmon, indica il processo sospetto, la tecnica ATT&CK candidata e tre query KQL di verifica. Cita gli identificativi degli eventi usati. Se le evidenze non bastano, dichiaralo.» Con l'ultima istruzione il modello può astenersi quando mancano elementi, pratica raccomandata per limitare le risposte inventate.

Il prompt fa parte della configurazione del sistema. Va versionato, provato su casi noti e sottoposto a revisione come il codice dei playbook; prompt modificati liberamente dai singoli analisti producono triage non riproducibili, con conseguenze anche sul piano probatorio (capitolo 8).

6.5 Allucinazioni dei modelli e loro misura

Un modello linguistico genera testo scegliendo, token dopo token, continuazioni probabili dato il contesto, senza verificarne la correttezza. Quando il contesto non contiene la risposta, il modello la completa con schemi appresi in addestramento. Il profilo NIST AI 600-1 classifica questo fenomeno tra i rischi tipici dell'AI generativa con il termine confabulation. Nel SOC ne derivano indicatori mai osservati, identificativi CVE inesistenti ma plausibili, comandi di remediation errati con sintassi corretta.

La contromisura consiste nel misurare la **groundedness**: ogni affermazione dell'output deve essere riconducibile a un'evidenza presente nel contesto, e ciò che non è riconducibile si tratta come ipotesi da verificare. La misura può essere manuale, su un campione di output, oppure automatica, confrontando le entità citate (indirizzi, hash, nomi di host) con quelle presenti nei dati forniti.

Scenario didattico. Chiesto di riassumere un incidente, il copilota aggiunge che l'host ha contattato un indirizzo IP associato alla rete Tor. L'indirizzo non compare in alcun log: riproduce uno schema frequente nei report usati per l'addestramento. Se l'indicatore entra nel report finale, nelle indagini successive viene trattato come un fatto accertato.

La regola operativa prevede che ogni indicatore, hash o CVE presente nell'output sia cercato nei dati originali prima dell'uso, e che i riassunti destinati a report formali restino bozze finché non passano dalla revisione umana con le fonti a fronte.

6.6 Modello locale o servizio cloud

Ogni prompt inviato a un servizio esterno è una **comunicazione di dati** al fornitore, e un trasferimento ai sensi del capo V GDPR se il fornitore li tratta fuori dallo Spazio economico europeo: log con indirizzi interni, nomi utente, estratti di posta elettronica, talvolta frammenti di malware. La scelta tra un modello in cloud e un modello eseguito in locale è una decisione di governance. Il modello in cloud è in genere più capace e non richiede gestione. Un modello locale ha capacità minori e costi infrastrutturali, ma lascia i dati sotto il controllo dell'organizzazione.

La decisione richiede la classificazione dei dati che possono uscire dall'organizzazione, contratti e accordi sul trattamento dei dati personali con il fornitore, informazioni sulla conservazione dei prompt presso il provider e sul loro eventuale uso per l'addestramento. Un'architettura ibrida assegna i dati d'indagine a un modello locale e riserva il cloud a compiti su dati pubblici o anonimizzati.

Scenario didattico. Un analista incolla in un chatbot pubblico il testo di una e-mail di spear phishing per farsela spiegare. Il testo contiene nome, ruolo e numero diretto del dirigente bersaglio, che finiscono nei sistemi di un fornitore situato fuori dallo Spazio economico europeo. Si configura un nuovo incidente, con possibili obblighi ai sensi dell'art. 33 GDPR.

Le misure di base sono una policy scritta su quali dati possono uscire, la pseudonimizzazione automatica dei prompt, l'uso esclusivo di modelli locali per i dati d'indagine forense e la conservazione del registro dei prompt come parte dell'audit trail del SOC.

6.7 Prompt, RAG e fine-tuning a confronto

Per adattare un modello linguistico alle esigenze di un SOC esistono tre leve, da considerare in quest'ordine:

- il **prompt engineering** (istruzioni, esempi, formato), che ha costi contenuti e si modifica senza riaddestrare il modello, purché ogni modifica sia versionata e provata (§ 6.4);
- la **retrieval-augmented generation** (RAG), che aggiunge la conoscenza dell'organizzazione senza modificare il modello: i documenti vengono indicizzati, i passaggi pertinenti recuperati a ogni domanda e inseriti nel contesto; l'indice si aggiorna senza riaddestramento e le risposte possono citare le fonti (capitolo 14);

- il **fine-tuning**, che modifica i pesi del modello con un addestramento ulteriore su dati specifici.

Il fine-tuning è utile per formati di output molto specifici o per il lessico di un dominio, ma non per fornire al modello informazioni fattuali aggiornate, compito per cui la RAG è più appropriata. Richiede inoltre dati curati, risorse di calcolo e manutenzione, fissa la conoscenza alla data dell'addestramento e rende più difficile stabilire che cosa il modello abbia appreso.

Scenario didattico. L'obiettivo sono note di triage uniformi, in italiano, con campi fissi. Un prompt con tre esempi copre la maggior parte dei casi; l'aggiunta della RAG sui playbook fa sì che le note citino la procedura corretta. Il fine-tuning, valutato, viene scartato perché il miglioramento atteso non compensa il costo di riaddestrare il modello a ogni modifica delle procedure.

Il fine-tuning su dati di indagini effettive incorpora quei dati nei pesi, e il modello può riprodurli: la distribuzione del modello può quindi comportare la comunicazione di quei dati (capitolo 15). Secondo il parere 28/2024 dell'EDPB, un modello addestrato su dati personali non può essere considerato anonimo in ogni caso, e l'anonimato va valutato caso per caso. La RAG, che mantiene i dati fuori dal modello, semplifica cancellazioni e controllo degli accessi.

6.8 Strumenti e permessi di un agente

Un **agente** è un modello linguistico inserito in un ciclo: osserva, pianifica, chiama uno strumento (una query al SIEM, una verifica di reputazione, l'apertura di un ticket), osserva il risultato e prosegue. Gli strumenti concessi delimitano ciò che l'agente può fare, e quindi i danni che un'allucinazione o una prompt injection possono provocare attraverso di esso. La OWASP Top 10 for LLM Applications 2025 classifica questo rischio come Excessive Agency (LLM06).

Le regole di progetto sono le seguenti:

- privilegio minimo per ogni strumento: l'agente di triage legge il SIEM ma non vi scrive;
- separazione tra lettura e azione, con le azioni subordinate ad approvazione (capitolo 5);
- account di servizio dedicati e tracciabili, mai credenziali personali;
- limiti per esecuzione (numero di chiamate, tempo, costo) che interrompono i cicli;
- registrazione di ogni chiamata con input e output.

Scenario didattico. Si consideri un agente di arricchimento degli alert che dispone di tre strumenti: ricerca nel SIEM in sola lettura, reputazione di indirizzi e domini, consultazione dell'inventario degli asset. Per ogni alert produce una scheda con le evidenze raccolte e una proposta motivata, senza poter chiudere l'alert né bloccare risorse, e un limite di venti chiamate per alert ne interrompe i cicli.

Un metodo di progetto consiste nel valutare, per ogni strumento concesso, che cosa potrebbe fare l'agente se un attaccante controllasse interamente il testo che legge. Se la valutazione indica un danno rilevante, lo strumento va limitato o subordinato ad approvazione.

6.9 Casi d'uso dei modelli linguistici nel SOC

Il modello è utile quando produce bozze che un esperto verifica in meno tempo di quanto impiegherebbe a scriverle. Si prestano a questo uso la sintesi e la riformulazione (riassunto di un caso per il passaggio di turno, versione per il vertice di un report tecnico), la generazione di query e regole da descrizioni in linguaggio naturale purché seguita da test, la prima stesura di report e timeline, la spiegazione di comandi e script offuscati, la consultazione della documentazione interna tramite RAG.

Presentano invece rischi elevati le decisioni autonome di contenimento, i giudizi definitivi sulla natura benigna o malevola di un oggetto senza verifica, i calcoli e le statistiche, e ogni compito in cui un errore passerebbe inosservato.

Scenario didattico. Al passaggio di turno il copilota riassume le ore precedenti a partire da ticket e chat operative: casi aperti, azioni svolte, attività in sospeso. Il turno entrante verifica il riassunto scorrendo i casi citati, con un costo basso rispetto al tempo risparmiato, anche se il rischio di errore resta.

Chiedere al modello se un allegato è malevolo, senza sandbox né analisi, produce risposte formulate con sicurezza in entrambe le direzioni e difficili da controllare. Il verdetto si costruisce con gli strumenti di analisi dei capitoli 9 e 11; al modello si può affidare la spiegazione del risultato.

6.10 Laboratorio: triage e detection con un LLM locale

Il modello usato è a pesi aperti ed è eseguito in locale con Ollama, così che nessun dato dell'esercitazione lasci la macchina dello studente. L'esercitazione comprende:

1. triage guidato di un alert esportato dal SIEM di laboratorio, con un prompt strutturato secondo il § 6.4;
2. generazione di una regola Sigma per la tecnica individuata;
3. conversione con sigma-cli e verifica su log di test, con un attacco simulato e traffico benigno;
4. confronto critico tra i contributi utili del modello e le allucinazioni riscontrate.

Consegna: la regola validata con gli esiti dei test; il registro delle allucinazioni riscontrate, con la frase del modello e il dato che la smentisce; i prompt usati, con una nota su ciò che ha funzionato. Il registro delle allucinazioni abitua a confrontare ogni affermazione del modello con i dati.

6.11 Letture

- NIST, *AI 600-1, Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile*, 2024.
- OWASP, *Top 10 for LLM Applications*, versione 2025.
- OWASP, *LLM AI Cybersecurity & Governance Checklist*, 2024.
- P. Lewis et al., «Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks», *Advances in Neural Information Processing Systems* 33, 2020.
- SigmaHQ, specifica Sigma e documentazione di pySigma e sigma-cli.
- Documentazione di Microsoft Security Copilot.

6.12 Domande di autovalutazione

1. Cosa significa grounding di un copilota sui dati del tenant? *Traccia.* Significa che il sistema recupera i dati dell'organizzazione (log, alert, identità) attraverso i connettori e li inserisce nel contesto del modello, così che la risposta si basi su quelle evidenze. Il grounding riduce le allucinazioni senza eliminarle, per cui la groundedness va misurata.
2. Come si valida una regola generata da un LLM prima del deployment? *Traccia.* Si converte la regola per il SIEM di destinazione, la si prova su log di un attacco simulato (deve rilevarlo) e su traffico benigno rappresentativo (si misurano i falsi positivi), si controllano i nomi dei campi rispetto allo schema reale dei log, poi la si sottopone a revisione tra pari e la si versiona.
3. Quali dati non dovrebbero transitare verso un LLM cloud? *Traccia.* Dati personali non pseudonimizzati, dati d'indagine forense, credenziali e segreti, informazioni classificate o coperte da obblighi contrattuali, salvo accordi specifici con il fornitore. La decisione deriva dalla classificazione dei dati e da una policy scritta.
4. Quali compiti di livello 1 sono candidati all'autonomia agentica? *Traccia.* Deduplicazione degli alert, raccolta del contesto, chiusura motivata di falsi positivi ricorrenti noti, con strumenti in sola lettura, controllo a campione degli esiti e registrazione completa. Le azioni di contenimento restano subordinate ad approvazione.
5. Come si misura l'allucinazione in un output investigativo? *Traccia.* Verificando che ogni affermazione e ogni entità citata (indirizzi, hash, host, CVE) sia presente nelle evidenze fornite al modello. La misura può essere manuale su un campione o automatica, confrontando le entità dell'output con quelle dei dati di partenza.

Capitolo 7. Fondamenti di Digital Forensics

Il capitolo stabilisce il metodo della digital forensics, comune a tutti i percorsi e sviluppato dalla specializzazione Digital Forensics nei capitoli 8–11. Descrive le fasi del processo forense secondo gli standard ISO/IEC, le principali fonti di evidenza nei file system e nel sistema operativo Windows e le regole processuali italiane che determinano l'utilizzabilità della prova digitale. Gli approfondimenti riguardano l'integrità della copia forense, la catena di custodia, l'ordine di volatilità, gli artefatti di esecuzione e di attività, le tecniche anti-forensi e il ruolo dell'esperto nel processo. Il capitolo privilegia il rigore sulla rapidità, in contrasto voluto con la logica di risposta del capitolo 1: le scelte di contenimento discusse lì vanno rilette alla luce dei vincoli qui descritti, e nella specializzazione Digital Forensics il capitolo 8 valuterà gli strumenti di AI rispetto agli stessi requisiti.

Percorso. Traccia core, in apertura del corso e prima della scelta della specializzazione.

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- eseguire un'acquisizione forense correttamente documentata, con verifica dell'integrità;
- mantenere e verificare la catena di custodia di un'evidenza digitale;
- interpretare le strutture dei file system e gli artefatti di Windows per ricostruire l'attività di un utente.

7.1 Il processo forense in cinque fasi

La digital forensics applica metodi scientifici all'identificazione, all'acquisizione, alla conservazione, all'analisi e alla presentazione delle evidenze digitali, in modo che i risultati possano essere verificati da terzi e utilizzati in sede giudiziaria. La famiglia di standard ISO/IEC ne descrive le attività. La ISO/IEC 27037:2012 riguarda identificazione, raccolta, acquisizione e conservazione, e la ISO/IEC 27041:2015 la garanzia dell'idoneità dei metodi. La ISO/IEC 27042:2015 tratta l'analisi e l'interpretazione, mentre la ISO/IEC 27043:2015 definisce i principi e i processi dell'indagine, compreso il processo di presentazione.

A fini didattici il processo si articola in cinque fasi:

- **identificazione:** individuazione dei supporti e dei dati potenzialmente rilevanti (dischi, dispositivi mobili, account cloud, log);
- **acquisizione:** creazione di una copia forense con write blocker e calcolo dell'hash prima e dopo la copia;
- **preservazione:** conservazione del reperto e della copia, con catena di custodia, sigilli e registro degli accessi;
- **analisi:** esame della copia, mai dell'originale, con strumenti e metodi documentati;
- **presentazione:** relazione tecnica e, se richiesto, esame in udienza.

La ISO/IEC 27037 distingue la raccolta, cioè la presa in carico del supporto fisico, dall'acquisizione, cioè la creazione di una copia dei dati. L'acquisizione può essere fisica, quando copia bit per bit l'intero supporto compreso lo spazio non allocato, oppure logica, quando copia file, cartelle o partizioni selezionati. La copia logica è più rapida e talvolta l'unica possibile (capitolo 10), ma conserva meno informazioni di quella fisica. I formati più usati sono l'immagine raw (prodotta per esempio con dd), il formato E01 (Expert Witness Format), che incorpora metadati e hash, e AFF4.

Il processo si fonda su alcuni requisiti di metodo. La **non alterazione** richiede che l'originale non venga modificato e che ogni accesso sia mediato e documentato.

La **ripetibilità** richiede che, con gli stessi dati, lo stesso metodo e lo stesso ambiente di prova, si ottenga lo stesso risultato. Per la **riproducibilità**, che la ISO/IEC 27037 tratta come requisito distinto, lo stesso risultato deve ottenersi in un ambiente diverso, anche da parte di un terzo. L'hash calcolato all'acquisizione e verificato dopo l'analisi documenta l'integrità della copia su cui il risultato si basa.

7.2 Dove si trovano le evidenze

Le evidenze digitali si trovano nelle strutture del file system, nei registri e nei log del sistema operativo e nello spazio non allocato. Il file system NTFS, usato da Windows, registra ogni file in un record della Master File Table (MFT), con i timestamp di creazione, modifica, accesso e modifica del record; il journal \$LogFile e il journal delle modifiche USN (\$UsnJrnl) annotano le operazioni sul file system. ext4, il file system più diffuso su Linux, organizza i metadati in inode e mantiene un journal; APFS, usato da Apple, adotta il copy-on-write, supporta snapshot e integra la cifratura.

Nei sistemi Windows le fonti più usate sono:

- **MFT e timestamp**: esistenza, creazione e modifica dei file;
- **Prefetch**: esecuzione di un programma, numero di esecuzioni e ultime date di avvio (cartella C:\Windows\Prefetch; il prefetching è attivo per impostazione predefinita sulle versioni client di Windows, mentre di norma è disattivato sulle edizioni Server e, in Windows 7, sulle unità SSD con buone prestazioni);
- **registro di sistema**: dispositivi USB collegati, attività dell'utente, programmi usati di recente (hive SYSTEM, SOFTWARE, NTUSER.DAT);
- **shellbag**: cartelle aperte dall'utente, anche se poi cancellate o presenti su supporti rimossi (NTUSER.DAT e UsrClass.dat);
- **registri eventi**: accessi, servizi, errori, cancellazione dei log;
- **spazio non allocato**: dati di file cancellati, recuperabili con tecniche di carving.

La cancellazione ordinaria rimuove il riferimento al file e lascia i dati sul supporto finché i settori non vengono sovrascritti. Il **carving** ricostruisce i file a partire dal contenuto, riconoscendo intestazioni e strutture tipiche dei formati. Sulle unità SSD il comando TRIM comunica al con-

troller i blocchi non più in uso, che possono essere cancellati in breve tempo, e il recupero di file cancellati da SSD è quindi spesso limitato.

7.3 La prova digitale nel processo

Nell'ordinamento italiano la Legge 18 marzo 2008, n. 48, che ha ratificato la Convenzione del Consiglio d'Europa sulla criminalità informatica (Budapest, 2001), ha modificato numerose disposizioni del codice di procedura penale. Negli artt. 244, 247 e 352 c.p.p. (ispezioni e perquisizioni) ha introdotto il richiamo all'adozione di misure tecniche dirette ad assicurare la conservazione dei dati originali e a impedirne l'alterazione. Nell'art. 354 c.p.p. ha previsto che negli accertamenti urgenti la polizia giudiziaria provveda, ove possibile, all'immediata duplicazione dei dati mediante una procedura che assicuri la conformità della copia all'originale e la sua immodificabilità. Con l'art. 254-bis c.p.p. ha disciplinato il sequestro di dati presso i fornitori di servizi, che può avvenire mediante copia con una procedura che assicuri la conformità dei dati acquisiti a quelli originali e la loro immodificabilità, fermo l'ordine al fornitore di conservare e proteggere adeguatamente i dati originali. Le norme non indicano strumenti specifici, e il rispetto del requisito si dimostra con il metodo e con la documentazione.

Il codice distingue gli accertamenti tecnici ripetibili, che il pubblico ministero può disporre affidandoli a un consulente (art. 359 c.p.p.), da quelli non ripetibili, per i quali l'art. 360 c.p.p. prevede garanzie per la persona sottoposta alle indagini, la persona offesa e i difensori (§ 7.9). Una copia forense eseguita correttamente e verificata con l'hash rende ripetibili le analisi successive, che possono essere rifatte da chiunque sulla stessa copia.

La **catena di custodia** documenta ogni passaggio del reperto: chi, che cosa, quando, dove e perché (§ 7.5). L'hash funziona come sigillo logico della copia, mentre il supporto originale è protetto dai sigilli fisici e dalla custodia controllata. Una lacuna nella catena non rende automaticamente inutilizzabile l'evidenza, ma la espone a contestazioni che il consulente dovrà affrontare in dibattimento. Gli stessi criteri saranno usati nel capitolo 8 per valutare l'ammissibilità dei risultati prodotti con strumenti di AI.

7.4 Funzioni hash e integrità della copia forense

Una funzione hash crittografica produce per ogni input un'impronta di lunghezza fissa. La modifica di un solo bit dell'input cambia l'impronta in modo imprevedibile, ed è computazionalmente impraticabile trovare un input con un'impronta data o due input diversi con la stessa impronta. Si raccomanda SHA-256; MD5 e SHA-1, ancora diffusi negli strumenti e nei verbali, sono vulnerabili a collisioni costruibili e vanno affiancati o sostituiti.

Calcolare l'hash al momento dell'acquisizione e verificarlo a ogni passaggio successivo dimostra che i dati analizzati sono identici a quelli acquisiti. Il **write blocker**, hardware o software, interviene prima: impedisce che il collegamento del supporto alla workstation di acquisizione ne modifichi il contenuto, per esempio aggiornando i timestamp di accesso o il journal del file system.

Scenario didattico. Un disco da 500 GB viene acquisito in formato E01. Lo SHA-256 calcolato sul dispositivo coincide con quello dei dati contenuti nell'immagine ed entrambi vengono riportati nel verbale. Sei mesi dopo, prima della consulenza di parte, il consulente ricalcola l'hash e ottiene lo stesso valore, quindi l'analisi si svolge su dati integri e chiunque può ripetere la verifica.

Un hash mancante o calcolato solo a posteriori espone a contestazioni. Anche un hash corretto attesta soltanto l'integrità della copia rispetto al momento del calcolo, e non dimostra la correttezza dell'acquisizione (per esempio la completezza della copia di un supporto con settori danneggiati) né la provenienza del dato.

7.5 Catena di custodia delle evidenze digitali

La catena di custodia è la documentazione ininterrotta della vita di un'evidenza: chi l'ha acquisita e con quali strumenti, dove è stata conservata, chi vi ha avuto accesso, per quale motivo e con quale esito. Con essa il giudice può escludere che l'evidenza sia stata alterata, sostituita o contaminata tra il sequestro e il dibattimento. Ogni passaggio di mano è registrato e firmato; un intervallo non documentato offre argomenti alla controparte.

La ISO/IEC 27037 descrive i ruoli coinvolti nelle prime fasi: il Digital Evidence First Responder (DEFR), autorizzato e formato per intervenire per primo sulla scena, e il Digital Evidence Specialist (DES), con competenze specialistiche. Il modulo di custodia riporta, per ogni reperto, descrizione e identificativi, hash, sigilli, date e firme dei passaggi.

Scenario didattico. Un notebook sequestrato passa dall'ufficiale di polizia giudiziaria al deposito dei corpi di reato e poi al consulente tecnico. Il modulo di custodia registra tre consegne con data, ora, firma e stato dei sigilli. In udienza la difesa chiede conto di un intervallo di due giorni: il registro del deposito, con il numero di protocollo, documenta dove si trovava il reperto.

Quando uno strumento di AI elabora un'evidenza, anche quel passaggio entra nella catena con versione del modello, parametri e hash dell'input e dell'output, e un'analisi automatica non registrata costituisce una lacuna (capitolo 14).

7.6 Ordine di volatilità e scelte di acquisizione

Le evidenze digitali hanno durate diverse. La RFC 3227 (2002) indica un ordine di acquisizione dalla fonte più volatile alla più stabile. In testa vengono registri e cache del processore, poi tabelle di routing, cache ARP, tabella dei processi, statistiche del kernel e memoria. Seguono i file system temporanei, il disco, i log remoti e i dati di monitoraggio, quindi la configurazione fisica e la topologia di rete e, per ultimi, i supporti d'archivio, come le copie di backup.

Acquisire nell'ordine sbagliato distrugge le evidenze più fragili, perché ogni azione sul sistema acceso consuma memoria e sovrascrive tracce. Da qui la scelta del primo intervento: acquisire i dati volatili a sistema acceso (**live forensics**), accettando di perturbarlo in misura documentata,

e poi il disco, oppure spegnere subito, preservando lo stato del disco e perdendo il contenuto volatile; anche la modalità di spegnimento, arresto ordinato o distacco dell'alimentazione, incide sulle tracce.

Scenario didattico. Un ransomware è in esecuzione su un server. In memoria possono trovarsi la chiave di cifratura in uso, le connessioni verso il server di comando e il processo malevolo. Se l'host viene spento senza acquisire la RAM, il disco resta integro ma chiave e connessioni vanno perse; con la cifratura del disco attiva (per esempio BitLocker) lo spegnimento può impedire anche l'accesso al disco senza la chiave di ripristino.

La scelta va fatta prima, nel playbook di primo intervento: quando procedere con l'acquisizione live (cifratura del disco attiva, malware residente in memoria) e quando spegnere (cancellazione distruttiva in corso). Ogni azione sul sistema acceso va annotata con l'orario, perché modifica ciò che si sta acquisendo. L'analisi della memoria acquisita è oggetto del capitolo 9.

7.7 Artefatti di esecuzione e di attività in Windows

Il sistema operativo registra molte tracce dell'attività degli utenti e dei programmi. La MFT conserva il record di un file anche dopo la cancellazione, finché il record non viene riutilizzato; il journal USN annota creazioni, modifiche e rinomine. Il registro conserva i dispositivi USB collegati, le reti usate e i programmi eseguiti o presenti nel sistema. Le singole fonti hanno significati diversi: UserAssist (NTUSER.DAT) documenta l'avvio di programmi dall'interfaccia grafica; ShimCache (valore AppCompatCache della hive SYSTEM) e Amcache (file hive Amcache.hve in C:\Windows\AppCompat\Programs) attestano la presenza di un eseguibile e, a seconda della versione di Windows, possono indicarne l'esecuzione, con eccezioni note. I file di Prefetch indicano quali eseguibili sono stati avviati, quante volte e in quali date recenti; le shellbag registrano le cartelle visualizzate.

Nessun artefatto da solo prova una ricostruzione, perché ciascuno ha eccezioni e condizioni di validità. Il valore probatorio sta nella concordanza di fonti indipendenti, perché falsificarle tutte senza lasciare incoerenze è molto più difficile.

Scenario didattico. Si sospetta che un dipendente in uscita abbia copiato dati riservati. Il registro mostra una chiavetta USB non dichiarata collegata in una certa data, e le shellbag indicano l'apertura della cartella «Clienti_2025» su quel volume. Negli stessi minuti Prefetch documenta l'esecuzione di uno strumento di archiviazione, mentre il journal USN del disco interno registra la creazione di un archivio compresso, poi cancellato.

L'estrazione e la normalizzazione degli artefatti sono operazioni meccaniche, svolte da strumenti come Plaso (capitolo 9). Un modello linguistico può aiutare a interpretare artefatti poco comuni e a suggerire dove cercare; il significato probatorio di un timestamp e le sue eccezioni restano materia di valutazione dell'esperto.

7.8 Tecniche anti-forensi e loro tracce

Chi prevede un'indagine può cercare di ostacolarla. Le tecniche anti-forensi più comuni sono la cancellazione sicura (wiping) di file o dello spazio libero, l'alterazione dei timestamp (timestamping), la cancellazione dei log, l'uso di contenitori cifrati e l'esecuzione di codice solo in memoria.

Queste tecniche lasciano a loro volta tracce. L'esecuzione di uno strumento di wiping può comparire in Prefetch, e Amcache ne attesta almeno la presenza sul sistema. Il timestamping produce spesso incoerenze tra i timestamp dell'attributo `$STANDARD_INFORMATION`, modificabile tramite le normali funzioni del sistema, e quelli dell'attributo `$FILE_NAME`, che molti strumenti non modificano. Su un disco magnetico uno spazio libero interamente azzerato è statisticamente anomalo rispetto a un uso normale; sulle unità SSD con TRIM attivo i blocchi liberati sono di norma restituiti come zeri anche senza wiping. Anche la cancellazione del registro Security di Windows lascia un evento (ID 1102), e un'assenza selettiva di dati in un intervallo significativo, documentata con rigore, va riportata nella relazione come fatto.

Scenario didattico. Un file decisivo risulta creato due anni prima dei fatti. Nella MFT, però, il timestamp di `$FILE_NAME` riporta una data diversa. Il journal USN registra la creazione effettiva e, pochi minuti dopo, la modifica dei timestamp (motivo `BASIC_INFO_CHANGE`), in corrispondenza dell'esecuzione dello strumento di manipolazione, documentata da Prefetch, e l'alterazione così accertata si riporta nella relazione come fatto autonomo.

Le conclusioni non devono andare oltre ciò che le evidenze sostengono. «Lo spazio libero è stato sovrascritto con uno strumento compatibile con X in data Y» è un'affermazione verificabile; «ha distrutto le prove del reato» è una conclusione che spetta ad altri. Gli strumenti di AI possono aiutare a individuare incoerenze statistiche su grandi volumi, per esempio nella distribuzione dei timestamp.

7.9 Consulenti tecnici e accertamenti non ripetibili

Nel processo penale l'esperto interviene come perito nominato dal giudice, come consulente tecnico del pubblico ministero (art. 359 c.p.p.) o come consulente tecnico di parte; nel processo civile come consulente tecnico d'ufficio (CTU) o consulente di parte (CTP). I ruoli hanno doveri diversi, ma la relazione tecnica deve in ogni caso esporre metodo, strumenti e limiti così che un terzo possa verificarli.

Sul piano processuale si distingue tra accertamento **ripetibile**, come l'analisi su una copia forense verificata con l'hash, e accertamento **non ripetibile**, che riguarda un oggetto il cui stato è soggetto a modificazione, anche per effetto dell'accertamento stesso, come l'acquisizione da un sistema acceso o l'estrazione di dati da un dispositivo che ne altera lo stato. Per gli accertamenti tecnici non ripetibili disposti dal pubblico ministero l'art. 360 c.p.p. prevede l'avviso alla persona sottoposta alle indagini, alla persona offesa e ai difensori, che possono nominare consulenti tecnici e assistere alle operazioni. Se la persona sottoposta alle indagini formula riserva di promuovere incidente probatorio, il pubblico ministero non procede, salvo che l'accertamento,

se differito, non possa più essere utilmente compiuto; se procede ugualmente fuori da questa ipotesi, i risultati non sono utilizzabili nel dibattito. Una qualificazione errata, cioè trattare come ripetibile un accertamento che non lo era, può compromettere l'utilizzabilità del risultato.

Scenario didattico. Uno smartphone acceso con cifratura attiva va acquisito subito, e l'estrazione ne modifica lo stato. È un accertamento non ripetibile, da svolgere con le garanzie dell'art. 360 c.p.p., verbale dettagliato, strumenti e versioni documentati e, se possibile, videoregistrazione delle operazioni, mentre le analisi successive sulla copia estratta, verificata con l'hash, sono ripetibili.

Un'analisi svolta con strumenti di AI su una copia forense è riproducibile solo se modello, versione e parametri sono fissati e documentati e se la variabilità degli output è controllata (capitolo 8). Uno strumento cloud aggiornato automaticamente impedisce di riprodurre il risultato con lo stesso modello. La copia resta disponibile e l'accertamento resta ripetibile ai sensi del codice, ma la verifica del risultato diventa più difficile, e per questo sono preferibili strumenti locali e versionati.

7.10 Laboratorio: ricostruzione dell'attività da un'immagine disco

L'intero processo si ripercorre su un'immagine disco didattica, con strumenti diffusi e gratuiti: FTK Imager per l'acquisizione e la verifica, Autopsy come interfaccia grafica per l'analisi e The Sleuth Kit come insieme di strumenti a riga di comando.

Il lavoro segue la sequenza del processo forense:

1. acquisizione dell'immagine disco didattica con FTK Imager e verifica degli hash calcolati all'acquisizione; calcolo dello SHA-256 dei dati acquisiti con uno strumento documentato (per esempio `ewfverify -d sha256` per le immagini E01);
2. compilazione del modulo di catena di custodia, passo per passo;
3. analisi in Autopsy di prefetch, registro, shellbag e file cancellati;
4. ricostruzione dell'attività dell'utente simulato in una timeline motivata, in cui ogni evento è collegato all'artefatto che lo documenta.

Consegna: modulo di custodia completo e coerente; timeline dell'attività con gli artefatti a supporto; breve relazione con metodo, strumenti, versioni e hash. Nella valutazione un modulo di custodia incompleto pesa più di un'analisi incompleta.

7.11 Letture

- ISO/IEC 27037:2012, *Guidelines for identification, collection, acquisition and preservation of digital evidence*.
- ISO/IEC 27042:2015, *Guidelines for the analysis and interpretation of digital evidence*.
- ISO/IEC 27043:2015, *Incident investigation principles and processes*.
- B. Carrier, *File System Forensic Analysis*, Addison-Wesley, 2005 (capitoli su NTFS).
- H. Carvey, *Windows Forensic Analysis Toolkit*, 4^a ed., Syngress, 2014.

- D. Brezinski, T. Killalea, *RFC 3227, Guidelines for Evidence Collection and Archiving*, IETF, 2002.
- NIST, *SP 800-86, Guide to Integrating Forensic Techniques into Incident Response*, 2006.
- Legge 18 marzo 2008, n. 48; codice di procedura penale, artt. 359 e 360.

7.12 Domande di autovalutazione

1. Quali requisiti deve soddisfare la catena di custodia? *Traccia.* Deve documentare senza interruzioni ogni passaggio del reperto e della copia: chi li ha presi in carico, quando, dove sono stati conservati, chi vi ha avuto accesso e perché, con firme, sigilli e hash. Serve a escludere alterazioni, sostituzioni o contaminazioni.
2. Perché l'hash si calcola all'acquisizione e si verifica dopo l'analisi? *Traccia.* Il valore calcolato all'acquisizione fissa lo stato dei dati; la verifica successiva dimostra che la copia analizzata è identica a quella acquisita e che l'analisi non l'ha modificata. Chiunque può ripetere il calcolo, il che rende verificabile l'integrità.
3. Che differenza c'è tra acquisizione fisica e logica? *Traccia.* L'acquisizione fisica copia bit per bit l'intero supporto, compreso lo spazio non allocato, dal quale si possono recuperare file cancellati; quella logica copia file, cartelle o partizioni selezionati. La seconda è più rapida e talvolta l'unica possibile, ma conserva meno informazioni.
4. Quali artefatti di Windows attestano l'esecuzione di un programma? *Traccia.* Prefetch (eseguibili avviati, numero e date recenti di esecuzione), UserAssist (programmi avviati dall'interfaccia grafica), Amcache e ShimCache (presenza e, con eccezioni, esecuzione), oltre ai registri eventi quando è attiva la registrazione dei processi. Le conclusioni si basano sulla concordanza di più fonti.
5. Cosa comporta la distinzione tra accertamenti ripetibili e non ripetibili? *Traccia.* Gli accertamenti non ripetibili riguardano oggetti il cui stato è soggetto a modificazione, anche per effetto dell'accertamento stesso: se disposti dal pubblico ministero richiedono le garanzie dell'art. 360 c.p.p. (avviso alla persona sottoposta alle indagini, alla persona offesa e ai difensori, facoltà di nominare consulenti tecnici e di assistere, possibile riserva di incidente probatorio). Gli accertamenti su copia forense verificata sono ripetibili e possono essere rifatti da altri consulenti sulla stessa copia.

Capitolo 8. Digital Forensics assistita dall'AI

Il capitolo esamina l'impiego dell'intelligenza artificiale nel processo forense descritto nel capitolo 7. Le tecniche considerate intervengono nel triage dei supporti e degli artefatti, nella ricerca nei contenuti sequestrati e nella classificazione di immagini e documenti. Il capitolo tratta poi le condizioni alle quali un risultato ottenuto con un modello può entrare in una relazione tecnica e reggere al contraddittorio. Gli approfondimenti riguardano l'ordinamento della coda di lavoro, la ricerca semantica, la spiegabilità degli esiti, gli hash di similarità, il rilevamento della manipolazione di immagini e l'organizzazione del laboratorio su un binario di scoperta e uno di validazione.

Percorso. Specializzazione Digital Forensics (capitoli 8–11).

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- impiegare strumenti di AI per ridurre i tempi di triage su grandi volumi di dati, documentandone l'uso;
- giudicare criticamente l'affidabilità di una classificazione automatica;
- argomentare le condizioni di ammissibilità di un'evidenza individuata con l'AI.

8.1 Il problema del volume

Un procedimento che coinvolge più dispositivi raggiunge spesso terabyte di dati e milioni di file, e l'esame integrale di ogni elemento non è compatibile con i tempi del procedimento. Il triage assistito dall'AI interviene prima dell'analisi approfondita: ordina supporti e artefatti in base alla probabilità che contengano elementi rilevanti.

Le funzioni automatizzate comprendono:

- **classificazione** di immagini e documenti per categoria (documenti contabili, fotografie di persone, schermate, materiale illecito);
- prioritizzazione dei supporti per probabilità di rilevanza;
- raggruppamento (clustering) delle comunicazioni e individuazione delle relazioni tra interlocutori;
- deduplicazione e filtraggio dei file noti.

Il filtraggio dei file noti è la tecnica più antica e non richiede modelli: si calcola l'hash di ogni file (§ 7.4) e lo si confronta con insiemi di riferimento, il più usato dei quali è quello della National Software Reference Library (NSRL) del NIST. La corrispondenza indica che il file appartiene a un software noto e non ne attesta l'innocuità, per cui l'esclusione dall'esame va motivata in relazione al caso.

Il triage probatorio privilegia il **richiamo** (recall), cioè la quota di elementi rilevanti effettivamente individuati, rispetto alla precisione, cioè la quota di elementi segnalati che si rivelano rilevanti. Un falso negativo può far perdere un'evidenza, mentre un falso positivo costa tempo di

revisione. Nella detection operativa (§ 2.4) il bilanciamento è diverso, perché un eccesso di falsi positivi satura gli analisti, e lo stesso tipo di classificatore richiede quindi soglie diverse nei due contesti.

8.2 Interrogare l'evidenza in linguaggio naturale

A differenza della ricerca per parole chiave, la ricerca semantica restituisce i passaggi il cui significato è vicino a quello della domanda: una richiesta su «pagamenti sospetti» individua messaggi in cui si parla di contanti, di fatture da non registrare o di somme da «sistemare», anche se la parola «pagamento» non compare. Il meccanismo è descritto nel § 8.5.

Le principali suite forensi commerciali integrano funzioni di questo tipo. In Magnet Axiom, per esempio, Magnet Copilot accetta domande in linguaggio naturale su conversazioni e cronologie di ricerca, con risposte che citano i dati del caso ed elaborazione sulla workstation dell'esaminatore; Magnet Review offre ricerca in linguaggio naturale e sintesi strutturate dei contenuti. Una sintesi indica all'esaminatore quali conversazioni leggere per prime, ma un riassunto prodotto da un modello linguistico può omettere passaggi o attribuire affermazioni alla persona sbagliata, e in relazione entrano perciò i messaggi originali, letti dall'esaminatore.

Il risultato semantico orienta la ricerca, e l'evidenza resta il file originale, verificato nel suo contesto. Il richiamo non è garantito: la ricerca semantica si affianca a quella per parole chiave e alle espressioni regolari (numeri di carta, codici IBAN, indirizzi di wallet), più affidabili quando l'oggetto cercato ha una forma definita. Ogni interrogazione, con l'esito e la versione del modello, va registrata nel log del caso per poter essere ripetuta.

Inviare dati sequestrati a un servizio esterno equivale infine a comunicarli a un terzo, con effetti sulla riservatezza dell'indagine, sulla protezione dei dati personali e sulla catena di custodia (§ 7.5). Per questo si preferiscono modelli eseguiti in locale, come nell'esercitazione del § 8.10.

8.3 Ammissibilità degli esiti di un modello

Un riferimento classico sull'ammissibilità della prova scientifica è la sentenza della Corte suprema degli Stati Uniti *Daubert v. Merrell Dow Pharmaceuticals* (1993), che affida al giudice il compito di vagliare l'affidabilità del metodo su cui si fonda la testimonianza dell'esperto. La sentenza indica criteri da applicare con flessibilità:

- **verificabilità:** il metodo può essere controllato ed è stato verificato empiricamente;
- sottoposizione a peer review e pubblicazione;
- tasso di errore noto o stimabile, ed esistenza di standard che regolano l'applicazione del metodo;
- accettazione generale nella comunità scientifica di riferimento.

Con la sentenza *Kumho Tire Co. v. Carmichael* (1999) la verifica è stata estesa alle competenze tecniche non strettamente scientifiche, tra le quali rientra gran parte della digital forensics. La regola 702 delle Federal Rules of Evidence, modificata nel 2000 per recepire questi principi e di

nuovo con effetto dal 1° dicembre 2023, richiede che la parte che introduce l'esperto dimostri al giudice, secondo il criterio del più probabile che non, che i requisiti della regola sono soddisfatti, tra cui dati sufficienti e metodi affidabili applicati in modo affidabile al caso. Nel processo penale italiano la valutazione della prova tecnica spetta al giudice, che la compie sulla base di perizie e consulenze; la Corte di cassazione ha indicato criteri per vagliare l'affidabilità del sapere scientifico, tra i quali l'ampiezza e la rigorosità degli studi che sorreggono la teoria, il grado di consenso che la tesi raccoglie nella comunità scientifica e l'indipendenza del soggetto che gestisce la ricerca, insieme alle finalità per le quali si muove (Cass. pen., sez. IV, 17 settembre 2010, n. 43786, Cozzini).

Applicati a un modello di machine learning, questi criteri hanno conseguenze operative. La relazione documenta la versione del modello, i parametri e, per quanto disponibili, i dati di addestramento e di validazione. Ogni esito prodotto dal modello è verificato dal perito, che ne risponde. Dati di addestramento non rappresentativi producono errori sistematici, contestabili in giudizio. La sezione metodologica della relazione dichiara in quali passaggi è stata usata l'AI e con quale funzione.

Sul piano regolatorio, il Reg. (UE) 2024/1689 (AI Act) colloca tra i sistemi ad alto rischio dell'allegato III (punto 6, lettera c) quelli destinati a essere usati dalle autorità di contrasto, o per loro conto, per valutare l'affidabilità degli elementi probatori nel corso delle indagini o dell'azione penale, nella misura in cui il loro uso è consentito dal diritto dell'Unione o nazionale. Dopo il Reg. (UE) 2026/1744 i relativi obblighi si applicano dal 2 dicembre 2027; obblighi e implicazioni sono discussi nel capitolo 15.

8.4 Triage assistito e ordine della coda di lavoro

L'aumento del volume dei dati è una delle principali cause di ritardo nella digital forensics: un caso può comprendere decine di dispositivi e terabyte di dati, e la letteratura documenta arretrati di analisi che durano mesi o anni (Quick e Choo, 2014). Il triage assistito dall'AI ordina la coda di lavoro. Stabilisce quali dispositivi, partizioni o file esaminare per primi e assegna a ciascun elemento un **punteggio di rilevanza** calcolato su più segnali: tipo di file, corrispondenza con hash noti, esito dei classificatori di contenuto, andamento temporale dell'attività.

Le decisioni sull'esame spettano all'esaminatore, che riceve dal modello una proposta di ordine. Gli elementi collocati in fondo alla coda restano nell'insieme acquisito, integro e a disposizione delle altre parti, e possono essere esaminati in seguito, perché il punteggio non comporta l'esclusione o la distruzione di alcun dato.

Scenario didattico. Si consideri un sequestro di 12 dispositivi per 8 TB complessivi. Il triage automatico segnala un notebook con un programma di cancellazione sicura installato e migliaia di file eliminati di recente, e un telefono con un'app di messaggistica cifrata usata nelle ore del fatto. Il laboratorio inizia da questi due dispositivi e prosegue con gli altri dieci, senza escluderne nessuno.

Il punteggio riflette i dati di addestramento: un modello tarato su casi di frode può sottovalutare artefatti tipici di altri reati. Un altro limite riguarda chi usa lo strumento: l'effetto di **tunnel vision** porta a concentrare l'attenzione sugli elementi in cima alla coda, e quelli in fondo rischiano di non essere mai esaminati pur essendo formalmente disponibili. Le contromisure sono organizzative: soglie minime di esame per ogni dispositivo, campionamento casuale degli elementi a basso punteggio, indicazione in relazione del perimetro effettivamente esaminato (§ 8.9).

8.5 Ricerca semantica nei corpora sequestrati

La ricerca per parole chiave fallisce con gergo, errori di battitura, lingue miste e allusioni. La ricerca semantica rappresenta ogni frammento di testo come un **embedding**, cioè un vettore numerico prodotto da un modello linguistico in modo che testi di significato simile corrispondano a vettori vicini. Anche la domanda dell'esaminatore viene trasformata in vettore, e il sistema restituisce i frammenti più vicini secondo una misura di somiglianza, di solito il coseno, anche in assenza di parole in comune. Chat, e-mail e documenti sequestrati possono così essere interrogati per temi («pagamenti in contanti non registrati», «incontri da tenere riservati»).

Il modello di embedding, la suddivisione dei testi in frammenti e il numero di risultati restituiti influiscono sugli esiti e vanno documentati. La ricerca ibrida combina il punteggio semantico con quello per parole chiave e riduce i risultati mancati da ciascuno dei due metodi.

Scenario didattico. In un corpus di 200.000 messaggi la parola chiave «droga» restituisce tre risultati non pertinenti. La query semantica «consegna di sostanze dietro pagamento» individua conversazioni su «la solita roba», «quella cosa di sabato», «il pacco del giardiniere», espressioni allusive che una lista di parole chiave non avrebbe previsto.

Sul piano del metodo la ricerca semantica è uno strumento di **scoperta**. L'ordinamento dei risultati non ha valore probatorio, i falsi negativi sono possibili e l'assenza di risultati non dimostra l'assenza di conversazioni sul tema. I risultati rilevanti vanno riscontrati sul dato originale, nel suo contesto completo: il messaggio precedente e quello successivo possono cambiare il senso di un'espressione allusiva.

8.6 Spiegabilità degli esiti in sede di controesame

Un risultato che l'esperto non sa spiegare è difficilmente difendibile in contraddittorio. Applicati al machine learning, i criteri Daubert (§ 8.3) portano in genere a chiedere conto dei dati di addestramento e di validazione del modello, delle metriche di errore, della riproducibilità dell'output, che richiede lo stesso risultato a parità di versione, parametri e input, e delle caratteristiche dell'input che hanno pesato sulla classificazione. Il **tasso di errore** va riferito a una popolazione definita: un classificatore validato su fotografie recenti da smartphone può rendere meno su immagini scansionate o ricomprese.

Le caratteristiche determinanti si indicano con le tecniche di **explainable AI**: SHAP e LIME stimano il contributo di ciascuna caratteristica a una singola previsione, mentre le mappe di salienza evidenziano le regioni di un'immagine che hanno inciso di più. Queste spiegazioni approssimano a posteriori il comportamento del modello e possono variare con piccole modifiche dell'input; in relazione vanno presentate insieme ai loro limiti.

Esempio. Domande tipiche del controesame riguardano il tasso documentato di falsi positivi del software, la popolazione su cui è stato validato e la possibilità di ottenere oggi lo stesso risultato. L'esperto risponde con il rapporto di validazione, la versione del modello indicata nel verbale e la mappa di salienza che indica le regioni dell'immagine che hanno inciso di più sulla classificazione, presentata con i limiti delle spiegazioni a posteriori.

La relazione descrive sia la segnalazione prodotta dal modello sia la verifica condotta dall'esperto con metodo tradizionale. Se il risultato del modello non si può riprodurre, per esempio perché il modello è stato aggiornato o i parametri non sono stati registrati, la sua difesa in contraddittorio diventa molto difficile, mentre resta difendibile la verifica condotta sul dato originale. La conservazione della versione e della configurazione dello strumento fa quindi parte dell'acquisizione, come l'hash della copia forense (§ 7.4). Con i servizi aggiornati di continuo dal fornitore la riproducibilità non è sotto il controllo del laboratorio, e la relazione deve dichiararlo.

8.7 Hash di similarità e fuzzy hashing

L'hash crittografico (§ 7.4) dà un esito binario: due file sono identici o diversi, e la modifica di un solo bit cambia completamente il valore. Le indagini richiedono spesso di misurare la somiglianza: un documento leggermente modificato, una variante di un malware, un'immagine ricompresa. A questo servono gli hash di similarità. **ssdeep** suddivide il file in frammenti delimitati da punti scelti in base al contenuto (context-triggered piecewise hashing), calcola una firma breve per ciascun frammento e confronta le firme di due file, restituendo un punteggio da 0 a 100; **TLSH** sintetizza la distribuzione di brevi sequenze di byte e restituisce una distanza, in cui valori bassi indicano file simili. Per le immagini si usano gli hash percettivi, che descrivono l'aspetto visivo e resistono a ridimensionamento e ricomprensione.

Gli hash percettivi servono anche a riconoscere contenuti illeciti già catalogati dopo modifiche: PhotoDNA, sviluppato nel 2009 da Microsoft con Hany Farid, è usato da piattaforme e forze di polizia per individuare materiale pedopornografico noto. Nell'analisi del malware le misure di similarità servono a raggruppare i campioni in famiglie (§ 11.8). Data la natura probabilistica di queste tecniche, in giudizio una corrispondenza di similarità vale come indizio da verificare con il confronto diretto dei contenuti e non come identificazione.

Scenario didattico. Il contratto «originale» prodotto in giudizio ha un hash SHA-256 diverso da quello della copia sequestrata, come atteso per un file salvato di nuovo. ssdeep, applicato al testo estratto dai due file, restituisce un punteggio di 96 su 100, cioè contenuti in larga parte comuni con modifiche limitate, e il confronto puntuale delle differenze individua la clausola alterata.

I file molto piccoli producono firme poco informative; i formati compressi, come i documenti Office o i PDF con flussi compressi, possono cambiare quasi interamente per una modifica minima del contenuto, per cui il confronto si esegue sul testo estratto. Un punteggio di somiglianza non ha un significato assoluto e dipende dall'algoritmo e dal tipo di file. La relazione indica algoritmo, versione, soglia adottata e tasso atteso di falsi riscontri (§ 8.6).

8.8 Rilevamento della manipolazione di immagini

Prima di stabilire se un'immagine è stata generata da un modello (capitolo 10), l'analisi forense verifica se è stata manipolata. L'**Error Level Analysis** (ELA) ricomprime l'immagine a una qualità nota e confronta il risultato con l'originale: le regioni con una storia di compressione diversa, come un oggetto incollato da un'altra foto, mostrano livelli di errore anomali. Le tabelle di quantizzazione e le tracce di doppia compressione JPEG danno indicazioni sulla storia del file. L'analisi copy-move individua porzioni duplicate nella stessa immagine, tipiche della cancellazione di un oggetto per clonazione dello sfondo, mentre le incoerenze di luce, ombre, prospettiva e rumore segnalano il fotomontaggio.

Il **PRNU** (Photo Response Non-Uniformity) è l'impronta di rumore che ogni sensore imprime sulle immagini per effetto di minime differenze di sensibilità tra i pixel. Stimato da un insieme di fotografie di riferimento, serve ad associare una foto a una specifica fotocamera o a rilevare porzioni non compatibili con quel sensore. I modelli di machine learning integrano questi segnali e li applicano a grandi raccolte; l'esperto deve saper spiegare in udienza i principi fisici su cui si basano.

Scenario didattico. In una controversia assicurativa la foto del danno presenta, nella zona del parafrangente, valori ELA anomali e un rumore incompatibile con il resto dell'immagine; l'analisi copy-move individua la stessa porzione di carrozzeria duplicata e ruotata. La manipolazione è sostenuta da tre segnali indipendenti, e la relazione riporta le mappe di analisi oltre alla conclusione.

Ogni tecnica ha falsi positivi noti. L'ELA segnala anche salvataggi multipli legittimi e regioni ricche di dettagli, e le piattaforme social ricomprimono le immagini cancellando molte tracce; il confronto PRNU richiede la fotocamera o un numero adeguato di immagini di riferimento. Una conclusione richiede la convergenza di più segnali indipendenti e la dichiarazione dei tassi di errore. Queste tecniche servono anche quando si sospetta un contenuto sintetico, che può essere a sua volta ritoccato o inserito in un fotomontaggio.

8.9 Doppio binario di scoperta e validazione

L'integrazione dell'AI nel flusso di laboratorio si organizza su un **doppio binario**. Il binario di scoperta comprende triage, ricerca semantica, hash di similarità e classificatori: strumenti rapidi e fallibili, i cui risultati sono piste. Il binario di validazione verifica ogni pista promossa con metodo tradizionale sul dato originale (esame dell'artefatto, ricalcolo dell'hash, ricostruzione del contesto) e alimenta la relazione finale.

Tra i due binari si tiene un registro, con cui il consulente spiega come ha selezionato il materiale esaminato: strumenti di AI usati con versione e parametri, dati su cui sono stati applicati, piste promosse o scartate con la relativa motivazione. La norma ISO/IEC 27041 indica come dimostrare che i metodi d'indagine, compresi gli strumenti che li realizzano, sono idonei allo scopo; uno strumento di AI va validato con i medesimi criteri, sui tipi di dati per cui viene usato.

Scenario didattico. Si supponga un caso da 6 TB: il binario di scoperta riduce il materiale a 300 file candidati in una giornata. Il binario di validazione li esamina in una settimana e ne porta 41 in relazione. Il registro intermedio documenta entrambi i passaggi e può essere esaminato dal consulente di parte.

Il materiale non promosso dalla fase di scoperta resta non esaminato, e i limiti del triage (§ 8.4) riducono il perimetro effettivamente esaminato. Le contromisure sono soglie prudenti, l'indicazione di tale perimetro in relazione e il campionamento casuale del materiale scartato con esame manuale del campione; se il campione rivela elementi rilevanti, la configurazione della fase di scoperta va rivista.

8.10 Laboratorio: confronto tra triage manuale e assistito

Il caso didattico comprende alcune migliaia di file, con un insieme di elementi rilevanti predefinito dal docente. Metà dei gruppi esegue un triage manuale per categorie e parole chiave; l'altra metà usa la classificazione automatica e la ricerca semantica con un modello eseguito in locale. Ogni gruppo misura il tempo impiegato e il richiamo (§ 8.1) sugli elementi predefiniti.

Ogni gruppo redige poi la sezione metodologica della relazione e dichiara l'uso dell'AI secondo i criteri del § 8.3: strumento, versione, parametri, query, verifiche sul dato originale. Il confronto in plenaria conclude l'esercitazione, con attenzione agli elementi mancati da ciascun gruppo e alle loro cause.

Consegna: tabella di tempi e richiamo per approccio; elenco degli elementi mancati con la causa ipotizzata; sezione metodologica difendibile in contraddittorio.

8.11 Letture

- E. Casey, *Digital Evidence and Computer Crime*, 3ª ed., Academic Press, 2011, capitoli metodologici.
- *Daubert v. Merrell Dow Pharmaceuticals, Inc.*, 509 U.S. 579 (1993); *Kumho Tire Co. v. Carmichael*, 526 U.S. 137 (1999).

- ENFSI, *Best Practice Manual for the Forensic Examination of Digital Technology*, 2015.
- ISO/IEC 27041:2015, *Guidance on assuring suitability and adequacy of incident investigative method*; ISO/IEC 27042:2015, *Guidelines for the analysis and interpretation of digital evidence*.
- D. Quick, K.-K. R. Choo, «Impacts of increasing volume of digital forensic data: A survey and future research challenges», *Digital Investigation*, 11(4), 2014, pp. 273–294.
- J. Kornblum, «Identifying almost identical files using context triggered piecewise hashing», *Digital Investigation*, 3 (suppl.), 2006.
- H. Farid, *Photo Forensics*, MIT Press, 2016.

8.12 Domande di autovalutazione

1. Quali criteri Daubert sono critici per un classificatore di machine learning? *Traccia*. Sono critici il tasso di errore, che deve essere noto e riferito a una popolazione definita, e la verificabilità, che richiede di riprodurre l'esito con la stessa versione e gli stessi parametri. Conta inoltre la validazione su dati rappresentativi del caso. Per i modelli commerciali poco documentati la relazione si appoggia al rapporto di validazione e alle verifiche sul dato originale.
2. Perché nel triage probatorio il richiamo conta più della precisione? *Traccia*. Un falso positivo costa tempo di revisione, mentre un falso negativo può escludere dall'esame un'evidenza rilevante. Nella detection operativa il bilanciamento è diverso perché i falsi positivi saturano gli analisti; nel triage si fissano soglie prudenti e si campiona il materiale scartato.
3. Come si documenta l'uso di un modello nella relazione tecnica? *Traccia*. Nella sezione metodologica: strumento e versione, parametri, dati su cui è stato applicato, query o configurazioni, metriche di errore note, verifiche eseguite sul dato originale, perimetro esaminato. Il registro tra scoperta e validazione documenta le piste promosse e scartate.
4. Quali bias può introdurre un classificatore addestrato su dati non rappresentativi? *Traccia*. Errori sistematici su categorie, lingue, dispositivi o formati poco presenti nei dati di addestramento, per esempio artefatti di reati diversi da quelli su cui il modello è stato tarato. A questi si aggiunge la tunnel vision indotta dall'ordinamento della coda. Le contromisure sono la validazione sui dati del caso, il campionamento del materiale scartato e la dichiarazione dei limiti.
5. In quali fasi del processo forense l'AI è oggi più matura? *Traccia*. Nelle attività di scoperta: deduplicazione, classificazione di immagini e documenti, ricerca semantica, raggruppamento delle comunicazioni. L'interpretazione degli artefatti e le conclusioni restano attività dell'esperto, che verifica ogni esito con metodo tradizionale e ne risponde.

Capitolo 9. Memory e network forensics con AI

Il capitolo tratta la memoria volatile e il traffico di rete, fonti di evidenza che negli incidenti in corso contengono informazioni assenti dal disco. Descrive l'acquisizione e l'analisi della memoria con Volatility 3, la costruzione di una super-timeline con Plaso e Timesketch e il rilevamento dei canali di comando e controllo nel traffico cifrato, indicando per ciascuna tecnica il contributo dei metodi statistici e di machine learning e i controlli che ne verificano gli esiti. Gli approfondimenti riguardano il malware fileless, la ricostruzione cronologica, il rilevamento del beaconing, gli aspetti pratici dell'acquisizione della memoria, l'analisi dei log DNS e la scelta dei livelli di visibilità della rete.

Percorso. Specializzazione Digital Forensics (capitoli 8–11). Letture di raccordo: § 3.6 e 5.9.

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- acquisire e analizzare un dump di memoria con Volatility 3;
- ricostruire una super-timeline e interrogarla con strumenti di analisi assistita;
- individuare pattern di beaconing verso server di comando e controllo con tecniche statistiche e di machine learning.

9.1 La memoria volatile come fonte di evidenze

Alcune informazioni esistono solo nella memoria di un sistema acceso e vanno perse allo spegnimento: i processi e i thread in esecuzione, con handle aperti e DLL caricate, le connessioni di rete attive e i socket, il codice iniettato in processi legittimi e il malware mai scritto su disco. In memoria si trovano anche credenziali, chiavi crittografiche e dati già decifrati, come il contenuto di un volume cifrato montato.

L'ordine di volatilità (§ 7.6) impone quindi di acquisire la RAM prima del disco. L'acquisizione si esegue a sistema acceso con strumenti dedicati, come WinPmem per Windows e AVML per Linux, o con le funzioni di raccolta delle piattaforme EDR. Lo strumento occupa memoria e avvia processi, e altera così lo stato del sistema: l'operazione va documentata come accertamento non ripetibile (§ 7.9).

L'analisi si svolge con **Volatility 3**, framework open source che ricostruisce le strutture del sistema operativo a partire dal dump. I plugin più usati elencano i processi seguendo le liste del kernel (pslist) o cercandone le strutture nell'intera memoria (psscan), individuano regioni che possono contenere codice iniettato (malfind), elencano connessioni e socket (netscan) ed estraggono moduli e regioni per l'analisi successiva. Il confronto tra pslist e psscan evidenzia i processi nascosti, presenti in memoria ma scollegati dalle liste del kernel.

Gli indicatori di iniezione più comuni sono le regioni con permessi di lettura, scrittura ed esecuzione (RWX) non associate a un file, i processi con un padre incoerente con la gerarchia attesa (per esempio lsass.exe non avviato da wininit.exe) e il **process hollowing**, nel quale l'im-

immagine di un processo legittimo viene sostituita in memoria con codice malevolo. L'AI contribuisce spiegando gli esiti dei plugin e ordinando i processi per grado di sospetto (§ 9.4); ogni segnalazione va verificata sull'artefatto.

9.2 La cronologia dell'attacco

La ricostruzione cronologica procede per livelli. Le sorgenti sono il file system, il registro di Windows, i log degli eventi, la cronologia dei browser e gli artefatti della memoria. **Plaso** estrae da queste sorgenti gli eventi datati e li normalizza in un formato comune, producendo milioni di righe anche per pochi host. In **Timesketch** più analisti consultano, filtrano e annotano queste righe, e gli analyzer automatici vi evidenziano anomalie e pattern noti. L'analista ricostruisce la sequenza dei fatti e la riconduce alle tattiche e tecniche di MITRE ATT&CK.

Gli analyzer e l'interrogazione assistita riducono milioni di righe a un numero gestibile di eventi candidati, e lasciano aperto il problema dei timestamp manipolati. Il **timestomping**, cioè la modifica deliberata delle date dei file da parte dell'attaccante, si rileva confrontando fonti indipendenti. In NTFS gli attributi \$STANDARD_INFORMATION e \$FILE_NAME della Master File Table conservano date distinte, e molti strumenti di timestomping modificano soltanto il primo: una discordanza tra i due, o tra le date di un file e gli eventi che ne documentano l'esecuzione, può indicare una manipolazione da confermare con altre fonti.

9.3 Rilevamento del C2 nel traffico cifrato

Il traffico di comando e controllo (C2) è in gran parte cifrato con TLS, e il suo contenuto non è osservabile senza intercettazione. Restano osservabili le caratteristiche del traffico, che costituiscono i segnali del **beaconing**, cioè dei contatti periodici dell'impianto malevolo con il server di controllo:

- periodicità: connessioni a intervalli regolari, con una variazione casuale (jitter) limitata;
- dimensioni dei pacchetti quasi costanti nel tempo;
- fingerprint TLS rari o incoerenti con il client dichiarato, calcolati di preferenza con JA4, perché la randomizzazione dell'ordine delle estensioni introdotta da Chrome all'inizio del 2023, e adottata da altri client, rende instabile il JA3;
- per il DNS tunneling, query lunghe, ad alta entropia e numerose verso uno stesso dominio.

Zeek produce metadati per ogni sessione (connessioni, DNS, HTTP, handshake TLS e, fino a TLS 1.2, certificati del server); Arkime indicizza i pacchetti catturati e li rende ricercabili. Sui metadati si calcola, per ogni coppia di host sorgente e destinazione, un punteggio di regolarità degli intervalli, e i modelli di machine learning analizzano le caratteristiche del traffico senza accedere al contenuto cifrato. Il limite principale riguarda i beacon lenti, con intervalli di ore o di giorni, e quelli mascherati nel traffico legittimo, per esempio con il domain fronting, che nasconde la destinazione reale dietro il dominio di un servizio ospitato sulla stessa rete di distribuzione dei contenuti.

Caso documentato. La backdoor SUNBURST è stata distribuita nel 2020 con gli aggiornamenti compromessi della piattaforma SolarWinds Orion. Secondo l'analisi pubblicata da FireEye nel dicembre 2020, restava inattiva fino a circa due settimane dopo l'installazione, contattava sottodomini generati di avsvmcloud[.]com e, nella fase successiva, mascherava il traffico C2 come comunicazione del protocollo Orion Improvement Program del prodotto. Poiché il traffico imitava quello del software legittimo, i singoli indicatori di rete, presi isolatamente, erano poco distintivi.

9.4 Memoria volatile e malware fileless

Molto malware recente è progettato per lasciare poche tracce su disco: risiede in memoria (fileless), si inietta in processi legittimi (process hollowing, DLL injection), cifra le proprie stringhe e le decifra solo durante l'esecuzione. Per essere eseguito, però, il codice deve trovarsi in chiaro in memoria, insieme alle chiavi in uso, alle connessioni di rete attive, alle credenziali in cache e alle righe di comando effettive. Volatility 3 analizza le strutture del sistema operativo, come le liste dei processi, le tabelle degli handle e le regioni di memoria mappate, e ne evidenzia le incoerenze: un processo svchost.exe con un padre diverso da services.exe, oppure regioni eseguibili non associate ad alcun file su disco.

Scenario didattico. Antivirus e scansione del disco non rilevano nulla, ma l'EDR segnala traffico anomalo. Il dump di memoria mostra nel processo explorer.exe una regione eseguibile con un beacon Cobalt Strike mai scritto su disco, da cui strumenti dedicati estraggono indirizzo del C2 e configurazione.

Un dump contiene centinaia di processi e migliaia di moduli e regioni di memoria. Modelli addestrati su dump benigni e malevoli possono ordinare le anomalie per grado di sospetto, e un modello linguistico può tradurre l'output dei plugin in ipotesi leggibili. La conferma resta manuale: estrazione della regione, calcolo dell'hash, analisi del campione (capitolo 11). L'analisi presuppone un'acquisizione corretta della RAM, trattata nel § 9.7.

9.5 Super-timeline e ricostruzione cronologica

Ogni fonte registra una parte della sequenza con il proprio orologio: file system (timestamp MACB, cioè di modifica del contenuto, accesso, modifica dei metadati e creazione), registro di Windows, log applicativi, cronologia del browser, eventi di autenticazione. Plaso, con gli strumenti log2timeline e psort, normalizza queste fonti in un'unica **super-timeline**, che spesso conta milioni di righe, e Timesketch la rende consultabile. A questa scala la lettura integrale non è praticabile. L'AI rileva anomalie negli intervalli, come un'attività di sistema alle 03:12 in un ufficio chiuso, raggruppa in episodi gli eventi correlati e risponde a interrogazioni in linguaggio naturale che verificano un'ipotesi, per esempio le esecuzioni di binari nuovi nei dieci minuti successivi a un accesso.

Scenario didattico. Una timeline di 4,2 milioni di eventi riguarda tre host. Il raggruppamento isola un episodio di undici minuti: accesso RDP da un indirizzo interno già compromesso, creazione di un servizio, esecuzione di rclone, cancellazione del servizio. L'esaminatore verifica i 40 eventi dell'episodio sulle fonti originali e li riporta in relazione.

Host con fusi orari diversi o orologi non sincronizzati producono sequenze apparenti errate. Il timestomping altera le date dei file (§ 9.2), e uno stesso campo ha significati diversi a seconda del file system e del sistema operativo. La normalizzazione in UTC, la registrazione dello scarto di ogni orologio rispetto a una fonte di riferimento e il confronto tra fonti indipendenti sono passaggi da documentare.

9.6 Rilevamento statistico del beaconing

Un impianto malevolo contatta periodicamente il server C2 per ricevere istruzioni. La cifratura nasconde il contenuto e lascia osservabili periodicità dei contatti, dimensioni dei pacchetti, durata delle sessioni e rapporto tra dati inviati e ricevuti. L'analisi statistica dei metadati raccolti da Zeek misura la regolarità degli intervalli tra connessioni successive, per esempio con la loro dispersione intorno al valore medio: un beacon con intervallo di 60 secondi e jitter del 10% produce una distribuzione stretta e riconoscibile. Strumenti open source come RITA calcolano punteggi di beaconing a partire dai log di Zeek. I modelli di machine learning combinano più indizi deboli: dominio registrato di recente, certificato autofirmato, fingerprint TLS raro, attività costante anche nelle ore notturne. Il certificato è osservabile solo nelle sessioni fino a TLS 1.2. In TLS 1.3 è cifrato, e restano visibili, salvo l'uso di Encrypted Client Hello, il nome del server nel ClientHello e il fingerprint del client.

Scenario didattico. Nei log Zeek di 30 giorni un host contatta lo stesso dominio in HTTPS ogni 300 secondi, con una variazione di circa 20 secondi, 24 ore su 24, con pacchetti di circa 1,2 KB. Nessuna firma si attiva e il dominio non compare nelle liste di reputazione. La regolarità anche notturna esclude la navigazione di un utente. Il traffico ha quindi le caratteristiche del beaconing e, se il contesto ne conferma la natura malevola, il primo contatto colloca nel tempo la compromissione.

Anche il software legittimo genera traffico periodico: aggiornamenti, telemetria, sincronizzazione dell'orologio, controllo della posta. Per qualificare il traffico come malevolo servono, oltre alla periodicità, il contesto (destinazione, data di inizio, processo sorgente) e la correlazione con memoria e timeline dello stesso host, cioè la convergenza delle tre fonti applicata nel laboratorio (§ 9.10).

9.7 Aspetti pratici dell'acquisizione della memoria

L'analisi descritta nel § 9.4 presuppone un dump eseguito correttamente, e l'acquisizione è la fase più fragile. Strumenti come WinPmem, AVML o la funzione di acquisizione dell'EDR vanno scelti e provati prima dell'emergenza, sui sistemi operativi effettivamente presenti. Il dump di una memoria in uso è soggetto a **smear**: le pagine cambiano durante la copia e alcune strutture

possono perdere coerenza. Il file di paging (pagefile.sys) e il file di ibernazione (hiberfil.sys) conservano pagine e stati precedenti della memoria e vanno acquisiti insieme al dump. Ogni acquisizione su sistema acceso va verbalizzata con strumento, versione, orario e hash del dump, perché modifica lo stato che misura (§ 7.9).

Nei sistemi virtualizzati lo snapshot dell'hypervisor è spesso l'acquisizione più pulita: congela memoria e disco in un istante coerente e non richiede agenti sul sistema osservato. Il file di memoria prodotto, dopo il calcolo dell'hash, si analizza con Volatility; alcuni formati richiedono i file di stato associati o una conversione preliminare.

Scenario didattico. Un server critico non può essere riavviato né sottoposto a un dump che ne rallenti il servizio. Lo snapshot della macchina virtuale eseguito dall'hypervisor non interviene sul sistema guest, e l'analisi prosegue su una copia.

Un errore frequente è installare lo strumento sul disco del sistema compromesso anziché avviarlo da un supporto esterno, con il risultato di sovrascrivere evidenze e di segnalare l'attività a un attaccante attivo. Ricorrono anche il salvataggio del dump su una condivisione di rete non protetta, l'omissione di pagefile e hiberfil e la mancata annotazione dell'ora esatta, necessaria per la correlazione con timeline e traffico. L'acquisizione avviata automaticamente da una piattaforma SOAR (capitolo 5), con strumenti e destinazioni predefiniti, riduce alcuni di questi errori.

9.8 DGA e DNS tunneling nei log DNS

I log DNS sono una fonte forense rilevante, perché quasi ogni attività di rete inizia con una risoluzione DNS e la loro conservazione ha costi contenuti. Con i **DGA** (Domain Generation Algorithm) il malware genera ogni giorno molti domini pseudocasuali e tenta di contattarli; l'operatore ne registra solo alcuni, e il blocco tramite liste diventa inefficace. La traccia statistica è costituita da sequenze di risposte NXDOMAIN, spesso verso nomi ad alta entropia. I classificatori basati su n-grammi di caratteri, entropia e pronunciabilità ottengono buoni risultati sui nomi pseudocasuali, che hanno distribuzioni di caratteri diverse da quelle dei domini scelti da persone. Fanno eccezione i DGA basati su liste di parole, che concatenano termini reali e producono nomi a bassa entropia: per questi contano soprattutto le sequenze di risposte NXDOMAIN e il comportamento dell'host.

Il **DNS tunneling** trasporta dati dentro query e risposte DNS, codificandoli nei sottodomini e nei record di risposta; strumenti come iodine e dnscat2 lo realizzano. Si riconosce da query lunghe e frequenti verso lo stesso dominio, sottodomini ad alta entropia, record TXT voluminosi e un volume di traffico DNS anomalo per l'host.

Scenario didattico. Nei log DNS di un mese un host produce circa 4.000 risposte NXDOMAIN al giorno per domini come xk2v9qp[.]net, comportamento tipico di un DGA. La data d'inizio delle sequenze colloca nel tempo l'infezione anche se il C2 contattato con successo non è più attivo; timeline e dump di memoria dello stesso host (§ 9.4 e § 9.5) confermano l'ipotesi e identificano il processo.

Anche servizi legittimi usano nomi generati: reti di distribuzione dei contenuti, telemetria, prodotti antivirus. Il DNS-over-HTTPS sposta inoltre le query fuori dalla vista del resolver aziendale, e la policy di rete che impone il resolver interno, prerequisito della visibilità, va decisa prima dell'incidente.

9.9 Livelli di visibilità della rete e conservazione

La visibilità di rete ha costi diversi secondo il livello di dettaglio. La **cattura integrale dei pacchetti** conserva tutto, compresi i file trasferiti, ma richiede molto spazio, pone problemi di riservatezza e, con il traffico TLS, rende disponibile poco contenuto; si conserva in genere per pochi giorni. I metadati di sessione di Zeek riportano origine, destinazione, volumi, protocollo, fingerprint TLS e, fino a TLS 1.2, il certificato. Occupano una frazione dello spazio, si conservano per mesi e sono la base delle analisi di beaconing e DGA. I record di **NetFlow** (o IPFIX) registrano volumi e coppie di indirizzi e possono essere conservati per anni. Una configurazione tipica combina i flow su tutta la rete, Zeek nei punti di transito principali e la cattura integrale attivabile durante un incidente.

Scenario didattico. Un'esfiltrazione viene scoperta sei mesi dopo. La cattura integrale è stata sovrascritta, ma i log Zeek mostrano 40 GB verso un indirizzo esterno in tre notti, con il fingerprint TLS di un client non usato in azienda. I file trasferiti non sono identificabili; volume, date e destinazione sono accertabili, e il fingerprint TLS indica il tipo di client usato, da confermare con gli artefatti dell'host.

Conservazione e punti di cattura si decidono in anticipo, bilanciando costi, esigenze d'indagine e proporzionalità. La cattura integrale del traffico dei dipendenti è un trattamento di dati personali che va giustificato alla luce del GDPR e dei limiti ai controlli a distanza posti dall'art. 4 della L. 20 maggio 1970, n. 300 (§ 15.8).

9.10 Laboratorio: ricostruzione di un'intrusione da tre fonti

L'esercitazione si basa su un caso d'intrusione simulato, per il quale sono disponibili il dump di memoria dell'host compromesso, i log Zeek della rete del laboratorio e gli artefatti per la timeline. Il lavoro sul caso comprende:

- analisi del dump con Volatility 3 per individuare il processo malevolo;
- correlazione con i log Zeek per identificare il canale C2;
- costruzione della super-timeline in Timesketch con gli analyzer automatici;

- ricostruzione della cronologia dell'attacco, ricondotta alle tecniche di MITRE ATT&CK, con le evidenze di ogni passaggio.

Le tre fonti devono convergere sulla stessa ricostruzione dei fatti. Una divergenza, per esempio un contatto C2 registrato in rete prima della creazione del processo che dovrebbe generarlo, va spiegata: uno scarto tra gli orologi, un secondo impianto, un errore di attribuzione.

Consegna: processo malevolo e canale C2 identificati, con le evidenze; timeline annotata in Timesketch; cronologia ATT&CK dell'intera intrusione.

9.11 Letture

- M. H. Ligh, A. Case, J. Levy, A. Walters, *The Art of Memory Forensics*, Wiley, 2014.
- Volatility Foundation, documentazione di Volatility 3.
- Documentazione di Plaso (log2timeline) e di Timesketch.
- D. Brezinski, T. Killalea, *RFC 3227, Guidelines for Evidence Collection and Archiving*, 2002.
- FireEye, *Highly Evasive Attacker Leverages SolarWinds Supply Chain to Compromise Multiple Global Victims With SUNBURST Backdoor*, dicembre 2020.
- X. Hu, J. Jang, M. Ph. Stoecklin, T. Wang, D. L. Schales, D. Kirat, J. R. Rao, «BAYWATCH: Robust Beaconing Detection to Identify Infected Hosts in Large-Scale Enterprise Networks», IEEE/IFIP DSN, 2016.
- D. Plohmann, K. Yakdan, M. Klatt, J. Bader, E. Gerhards-Padilla, «A Comprehensive Measurement Study of Domain Generating Malware», USENIX Security Symposium, 2016.

9.12 Domande di autovalutazione

1. Perché la memoria va acquisita prima dello spegnimento? *Traccia.* Processi, connessioni, codice iniettato, chiavi e dati decifrati esistono solo nella memoria del sistema acceso e vanno persi allo spegnimento. L'ordine di volatilità impone la RAM prima del disco; l'acquisizione altera lo stato del sistema e va verbalizzata come accertamento non ripetibile.
2. Quali indicatori in memoria rivelano una code injection? *Traccia.* Sono indicatori le regioni eseguibili, spesso RWX, non associate a file su disco (malfind) e i processi con un padre incoerente con la gerarchia attesa o con l'immagine sostituita (hollowing). Lo sono anche i processi visibili con psscan e assenti da pslist e le DLL caricate da percorsi insoliti. Ogni indicatore va confermato estraendo e analizzando la regione.
3. Che cosa aggiunge un analyzer di Timesketch rispetto alla timeline grezza? *Traccia.* Etichetta e raggruppa automaticamente eventi notevoli (anomalie temporali, pattern noti, episodi correlati) e riduce milioni di righe a un insieme di candidati. La conferma sulle fonti originali e il controllo del timestomping restano compiti dell'analista.
4. Quali caratteristiche statistiche distinguono il beaconing dal traffico legittimo? *Traccia.* Intervalli regolari con jitter limitato, dimensioni quasi costanti, attività continua anche notturna, destinazioni rare o recenti, fingerprint TLS inconsueti. Nessuna caratteristica basta da sola, perché anche il software legittimo è periodico: servono il contesto e la correlazione con memoria e timeline.

5. Quali limiti incontra la network forensics con il traffico cifrato? *Traccia.* Senza intercettazione TLS il contenuto non è osservabile, e in TLS 1.3 è cifrato anche il certificato del server, per cui l'analisi si basa sui metadati. Beacon lenti, domain fronting e traffico che imita protocolli legittimi riducono l'efficacia dei metodi statistici. Il DNS-over-HTTPS sottrae le query al resolver aziendale, e una conservazione breve della cattura integrale impedisce le verifiche a distanza di tempo.

Capitolo 10. Mobile, cloud e multimedia forensics

Il capitolo estende il metodo forense del capitolo 7 ai dispositivi mobili, agli ambienti cloud e ai contenuti multimediali, ambiti in cui il disco di un computer non è più la fonte principale. Per i dispositivi mobili descrive i livelli di acquisizione e il criterio di proporzionalità che guida la scelta. Per il cloud tratta l'indagine sui log dei fornitori a partire da un caso di compromissione della posta aziendale, e per i contenuti multimediali le tecniche di autenticazione e i limiti probatori dei detector di contenuti sintetici. Gli approfondimenti riguardano l'acquisizione mobile, le evidenze nel cloud, i deepfake e la provenienza dei contenuti, il recupero di dati dai database SQLite, le indagini su container e Kubernetes, i metadati e la geolocalizzazione.

Percorso. Specializzazione Digital Forensics (capitoli 8–11).

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- pianificare un'acquisizione mobile proporzionata al caso;
- condurre un'indagine sui log di ambienti cloud (Microsoft 365, AWS, Azure);
- applicare criticamente strumenti di rilevamento dei deepfake e di autenticazione multimediale.

10.1 Livelli di acquisizione e proporzionalità

L'acquisizione di un dispositivo mobile si colloca su tre livelli, con copertura e invasività crescenti:

- **logica:** dati esposti dalle interfacce del sistema e dai backup (contatti, messaggi, media); invasività minima, copertura parziale;
- **file system:** struttura dei file accessibile, compresi i database delle app, gli artefatti di messaggistica e i dati di geolocalizzazione;
- **fisica:** copia bit a bit della memoria, spazio non allocato incluso; cifratura e secure enclave la rendono spesso impraticabile.

La NIST SP 800-101r1 classifica invece gli strumenti in cinque livelli (estrazione manuale, estrazione logica, hex dumping e JTAG, chip-off, micro read); la distinzione tra acquisizione logica, del file system e fisica è quella d'uso negli strumenti commerciali. La cifratura attiva per impostazione predefinita su iOS e Android ha ridotto il ricorso all'acquisizione fisica a favore di quella logica, del file system e dei dati sincronizzati nei servizi cloud del produttore. Gli artefatti delle app di messaggistica, cioè database SQLite, allegati e metadati, sono spesso la fonte più rilevante.

La scelta del livello dipende dal caso e dalla base giuridica. È proporzionata l'acquisizione sufficiente agli scopi dell'indagine con la minore invasività: uno smartphone contiene la vita privata del titolare e dei suoi interlocutori, e un'estrazione integrale porta nel fascicolo molti dati estranei al reato. Le norme introdotte nel codice di procedura penale dalla L. 18 marzo 2008, n. 48 richiedono, per perquisizioni e sequestri di dati informatici, misure tecniche dirette ad assicu-

rare la conservazione dei dati originali e a impedirne l'alterazione; quando l'estrazione modifica il dispositivo, l'operazione rientra tra gli accertamenti tecnici non ripetibili (§ 7.9). La Corte di giustizia dell'Unione europea, con la sentenza del 4 ottobre 2024 nella causa C-548/21, ha stabilito che l'accesso della polizia ai dati di un telefono deve rispettare il principio di proporzionalità, tenendo conto della gravità del reato, ed essere sottoposto, salvo urgenza debitamente giustificata, al controllo preventivo di un giudice o di un'autorità amministrativa indipendente. L'acquisizione fisica e, sui dispositivi recenti, spesso anche quella completa del file system richiedono lo sfruttamento di vulnerabilità del dispositivo: in questi casi si aggiungono l'alterazione dello stato del dispositivo e l'onere di documentare strumento e metodo, la cui affidabilità può essere contestata.

10.2 Indagine BEC nei log del cloud

Nel **BEC** (Business Email Compromise) l'attaccante compromette o imita una casella di posta aziendale per deviare pagamenti, per esempio modificando le coordinate bancarie indicate in una fattura. Secondo i rapporti annuali dell'Internet Crime Complaint Center (IC3) dell'FBI, il BEC è tra le frodi informatiche con le perdite denunciate più elevate. L'indagine si svolge sui log del fornitore cloud:

- **Unified Audit Log** di Microsoft 365: accessi, creazione e modifica di regole di posta, accessi alle caselle;
- log di accesso di Microsoft Entra ID: provenienza geografica, dispositivi, esito dell'autenticazione a più fattori;
- AWS CloudTrail e Azure Activity Log: chiamate API, creazione di risorse e di chiavi;
- consensi OAuth concessi ad applicazioni di terze parti, che possono lasciare all'attaccante un accesso persistente anche dopo il cambio della password.

Il metodo cambia rispetto all'acquisizione di un supporto. Si acquisisce l'esportazione dei log, con hash e verbale di estrazione che indica account usato, filtri e intervallo temporale. I periodi di conservazione sono limitati e dipendono dal servizio e dalla licenza. Con Audit (Standard) lo Unified Audit Log conserva per 180 giorni i record generati dal 17 ottobre 2023, e con Audit (Premium) conserva per un anno quelli di Exchange, SharePoint, OneDrive ed Entra ID. I log di accesso di Entra ID restano disponibili per 7 giorni con Entra ID Free e per 30 giorni con P1 e P2, salvo esportazione verso un archivio esterno. La tempestività diventa quindi parte della custodia. Il fornitore è un soggetto terzo, e contratti e giurisdizione determinano quali dati si possono ottenere e in quali tempi. L'AI aiuta a correlare migliaia di eventi di accesso, per esempio raggruppandoli per indirizzo, dispositivo e orario e segnalando le combinazioni mai osservate per un utente; le correlazioni proposte vanno verificate sugli eventi originali.

10.3 Autenticità dei contenuti e AI generativa

L'analisi classica dei contenuti esamina metadati, livelli di errore di compressione e incoerenze di illuminazione e prospettiva (§ 8.8). I detector basati su machine learning sono addestrati a riconoscere volti, voci e video sintetici. Lo standard **C2PA** (Coalition for Content Provenance

and Authenticity) associa al file un manifesto firmato crittograficamente, che attesta il dispositivo o il software d'origine e le modifiche dichiarate; la firma ne prova l'integrità e lascia aperta la questione della veridicità della scena rappresentata.

I tassi di errore dei detector sono non trascurabili e più alti sui contenuti prodotti da generatori assenti dai dati di addestramento; le prestazioni si riducono sui contenuti ricompresi da social network e app di messaggistica, che sono una forma frequente in cui i contenuti arrivano all'indagine. L'esito di un detector non costituisce da solo una prova: la conclusione richiede la convergenza con metadati, contesto e testimonianze. Anche la provenienza ha limiti: molti dispositivi e piattaforme non generano il manifesto o lo rimuovono al caricamento, per cui la sua assenza non indica che il contenuto sia falso.

L'art. 50 del Reg. (UE) 2024/1689 (AI Act), applicabile dal 2 agosto 2026, impone ai fornitori di sistemi che generano contenuti sintetici di marcarli in un formato leggibile da una macchina (par. 2; per i sistemi già immessi sul mercato a quella data il Reg. (UE) 2026/1744 fissa l'adeguamento al 2 dicembre 2026). Impone inoltre ai deployer, cioè a chi usa il sistema al di fuori di un'attività personale non professionale, di rendere noto che immagini, audio o video costituenti un deep fake sono stati generati o manipolati artificialmente (par. 4), salvo le eccezioni previste. La marcatura facilita la verifica, ma può essere rimossa, e la sua assenza non prova l'autenticità. L'uso dei contenuti sintetici come strumento di frode è trattato nel § 12.5.

10.4 Livelli di acquisizione dei dispositivi mobili

Nei dispositivi mobili l'immagine completa della memoria è un'eccezione: la cifratura per file è attiva per impostazione predefinita, le chiavi sono protette da componenti hardware dedicati (secure enclave) e i bootloader sono bloccati. L'acquisizione **logica** raccoglie ciò che il sistema espone tramite backup e interfacce. Copiando l'albero dei file accessibile, l'acquisizione del **file system** include anche i database delle app, con i record cancellati ancora presenti negli spazi liberi e nei file WAL (§ 10.7). L'acquisizione fisica, che produce l'immagine della memoria flash, è oggi raramente possibile senza sfruttare vulnerabilità. La scelta è un compromesso documentato tra completezza, rischio di alterazione e base giuridica: ai livelli più profondi lo strumento interviene di più sul dispositivo.

Lo stato del dispositivo al momento del sequestro incide sull'esito. In un telefono acceso e sbloccato almeno una volta dopo l'avvio (stato AFU, After First Unlock) molte chiavi di cifratura sono disponibili in memoria; dopo un riavvio (stato BFU, Before First Unlock) la maggior parte dei dati resta inaccessibile senza il codice. Un dispositivo trovato acceso va quindi mantenuto acceso, alimentato e isolato dalla rete. Alcune versioni recenti dei sistemi operativi riavviano automaticamente il dispositivo dopo un periodo di inattività, riportandolo allo stato BFU: il tempo tra sequestro ed estrazione diventa un fattore da pianificare.

Esempio. Un'estrazione limitata ai dati esposti dalle interfacce restituisce i soli messaggi visibili nell'app; con l'accesso al file system si possono recuperare dal database SQLite della chat i record contrassegnati come eliminati e non ancora sovrascritti. La differenza tra un messaggio mai esistito e un messaggio cancellato può essere rilevante per il procedimento.

L'isolamento dalla rete (modalità aereo, contenitore schermato) evita cancellazioni da remoto. Ogni interazione con lo schermo può modificare lo stato del dispositivo e va verbalizzata: secondo l'impostazione della norma ISO/IEC 27037, quando un'alterazione non è evitabile, il verbale descrive ogni intervento e le sue conseguenze. Le funzioni di estrazione e analisi assistite dall'AI degli strumenti commerciali pongono gli stessi problemi di spiegabilità discussi nel § 8.6.

10.5 Evidenze negli ambienti cloud

Negli ambienti cloud l'investigatore non ha accesso fisico ai supporti. Le evidenze si trovano negli snapshot delle istanze e dei dischi virtuali, nei metadati dei servizi e nei **log del fornitore**, che registrano attività amministrative, accessi e chiamate API. Sono spesso replicate in giurisdizioni diverse e soggette a conservazione limitata, scaduta la quale i dati non esportati vengono cancellati. L'acquisizione richiede quindi tempestività: esportazione immediata dei log rilevanti con hash e verbale, documentazione delle configurazioni, richiesta formale di conservazione al fornitore. La catena di custodia riguarda soprattutto i processi: chi ha esportato, con quale account, con quali filtri, e come si dimostra la completezza dell'esportazione.

Scenario didattico. Si supponga un tenant Microsoft 365 compromesso con uno schema BEC. Lo Unified Audit Log mostra accessi dall'estero, la creazione di una regola di posta che inoltra le fatture a un indirizzo esterno e il consenso a un'applicazione OAuth malevola. Con una conservazione di 180 giorni, un giudizio civile successivo potrà basarsi sull'esportazione certificata nei primi giorni dell'indagine.

Per i dati conservati all'estero rilevano i trattati di assistenza giudiziaria, il CLOUD Act statunitense (2018) e la disciplina europea e-Evidence. Con il Reg. (UE) 2023/1543, che si applica dal 18 agosto 2026, l'autorità giudiziaria di uno Stato membro potrà rivolgere ordini europei di produzione e di conservazione direttamente ai fornitori di servizi che operano nell'Unione; la Dir. (UE) 2023/1544 obbliga i fornitori a designare stabilimenti o rappresentanti legali destinatari di tali ordini. Nei rapporti tra privati, il contratto con il fornitore determina quali dati il cliente può ottenere e in quali tempi.

10.6 Autenticazione dei contenuti e deepfake

Accanto alla domanda tradizionale sul ritocco di una foto si pone quella sull'esistenza stessa dell'evento rappresentato. Il rilevamento dei deepfake cerca le incoerenze lasciate dai generatori: illuminazione e riflessi negli occhi, anomalie nello spettro delle frequenze, artefatti di fusione e, nei video, dinamiche fisiologiche innaturali. Ogni nuova generazione di modelli riduce

però gli artefatti della precedente, e i detector perdono accuratezza sui contenuti ricompresi dalle piattaforme. Per questo si stanno diffondendo i sistemi di **provenienza**: lo standard C2PA (Content Credentials) firma crittograficamente l'origine del file e le modifiche registrate, e sposta la verifica sulla storia documentata del contenuto.

Caso documentato. Nel 2019, secondo quanto riferito dall'assicuratore Euler Hermes al Wall Street Journal, l'amministratore delegato della controllata britannica di un gruppo energetico autorizzò un bonifico di circa 220.000 euro dopo una telefonata in cui una voce sintetica imitava il vertice della capogruppo tedesca. Quando esiste una registrazione, l'analisi di casi simili combina spettrogramma e metadati del file con il contesto: tabulati, canale di arrivo, verifiche con le persone coinvolte.

I detector forniscono probabilità, da dichiarare con i tassi di errore (§ 8.6) e da confrontare con metadati, provenienza, testimonianze e tabulati. Esiste anche il **liar's dividend**, espressione introdotta da Chesney e Citron: la diffusione dei deepfake rende contestabile come falso anche un contenuto autentico.

10.7 Recupero di dati da database SQLite

Gran parte dei dati di interesse su un dispositivo mobile, come messaggi, chiamate, cronologie e posizioni, è conservata in database **SQLite**, nei quali i record cancellati non vengono rimossi subito. Un record eliminato finisce in spazio libero, cioè nei blocchi liberi delle pagine o nelle pagine della freelist, e resta leggibile finché non viene sovrascritto. Il **WAL** (write-ahead log) contiene le pagine modificate dalle transazioni recenti, mentre fino al checkpoint, cioè al trasferimento delle modifiche nel database, il file principale conserva le versioni precedenti. Anche i file di journal possono conservare copie di pagine anteriori alle modifiche. Le app possono però attivare l'opzione `secure_delete`, che sovrascrive con zeri i contenuti cancellati, oppure compattare il database con `VACUUM` o con l'impostazione `auto_vacuum`: in questi casi il recupero dal file principale si riduce fortemente o diventa impossibile, mentre copie delle pagine possono restare nel WAL, nei file di journal o nei backup.

Scenario didattico. Nell'ipotesi che l'indagato cancelli una conversazione la sera del fatto e che il telefono venga sequestrato acceso poche ore dopo, prima del checkpoint, il file principale del database conserva le pagine precedenti alla cancellazione, con i messaggi e i relativi timestamp, mentre il WAL contiene le pagine aggiornate. Anche la cancellazione, se collocabile nel tempo grazie ad altri artefatti, diventa un elemento di valutazione.

Il checkpoint avviene anche in automatico, per impostazione predefinita quando il WAL raggiunge le 1.000 pagine, e alla chiusura dell'ultima connessione al database, che elimina il WAL. L'analisi si svolge quindi su copie che comprendono i file `-wal` e `-shm`, perché aprire il file originale con un client SQLite ordinario può cancellare le versioni precedenti su cui si basa il recupero.

In relazione si distingue il dato presente nelle tabelle dal dato recuperato da spazio non allocato, perché affidabilità e contestabilità sono diverse. Ogni app usa uno schema proprio, che cambia tra una versione e l'altra: un modello linguistico può aiutare a interpretare schemi non documentati e a scrivere le query di estrazione. Il recupero da spazio non allocato resta affidato a strumenti dedicati e validati, e ogni risultato si verifica manualmente prima di entrare in relazione.

10.8 Indagini su container e Kubernetes

I container mettono in discussione un presupposto dell'analisi forense: che il sistema compromesso esista ancora quando l'indagine comincia. Un pod può restare attivo pochi minuti e, quando viene ricreato, riparte dallo stato dell'immagine: le modifiche dell'attaccante che non risiedono su volumi persistenti vanno perse. Le evidenze vanno quindi raccolte mentre il pod è attivo, con il checkpoint del container, la copia del layer scrivibile o lo snapshot del nodo, oppure devono essere già state esportate per scelta progettuale: log applicativi e **audit log** di Kubernetes inviati fuori dal cluster in tempo reale. In molte installazioni l'audit del control plane non è attivo per impostazione predefinita e va configurato.

L'indagine si basa sul confronto tra il container e la sua immagine, perché ogni differenza corrisponde ad attività svolta durante l'esecuzione, e sulla ricostruzione delle operazioni dai log centralizzati e dall'audit del control plane.

Scenario didattico. Si consideri un cryptominer attivo in un cluster, con il pod coinvolto già ricreato tre volte e quindi privo dello stato originale. L'audit log di Kubernetes mostra che il deployment è stato creato con le credenziali di un service account della pipeline CI, e il confronto tra l'immagine e il file system di un pod ancora attivo rivela il binario scaricato all'avvio.

Nei sistemi effimeri la **forensic readiness**, cioè la predisposizione alla raccolta delle evidenze prima dell'incidente, è un requisito di progetto: i dati non esportati in tempo reale possono non essere più disponibili. Lo stesso principio vale per gli ambienti cloud (§ 10.5) e per i sistemi di AI, in cui la registrazione degli eventi va progettata prima della messa in esercizio (§ 14.8).

10.9 Metadati e geolocalizzazione dei contenuti

Un file multimediale contiene più informazioni di quelle visibili. I dati **EXIF** di una foto possono includere coordinate GPS, data, modello della fotocamera e direzione dello scatto, e i video contengono tag analoghi. I documenti Office registrano autore, data di creazione e numero di revisione. I metadati si modificano con facilità e vanno confermati da altre fonti; le piattaforme sociali li rimuovono spesso al caricamento, per cui la copia prelevata dal dispositivo ha più valore di quella scaricata da una piattaforma.

In assenza di metadati si ricorre alla geolocalizzazione dal contenuto, che esamina ombre, vegetazione, cartelli e profilo degli edifici e del paesaggio. Questa tecnica di verifica OSINT è oggi supportata da modelli di visione che propongono luoghi candidati. Ogni luogo proposto è un'i-

potessi da verificare su immagini satellitari, fotografie d'archivio delle strade e fonti locali.

Scenario didattico. Un alibi si fonda su una foto che sarebbe stata scattata in montagna alle 17:40 del giorno X. Le coordinate EXIF sono coerenti, ma l'ombra indica un sole troppo alto per quell'orario; il modello digitale del terreno e le effemeridi solari collocano lo scatto intorno a mezzogiorno. Resta da stabilire se l'orario nei metadati sia stato alterato o derivi da un errore dell'orologio del dispositivo.

Gli strumenti di geolocalizzazione possono servire anche a localizzare persone senza il loro consenso; i profili di protezione dei dati e di proporzionalità sono trattati nel capitolo 15.

10.10 Laboratorio: indagine BEC e valutazione dei detector

L'esercitazione si articola in due parti. La parte A riguarda un caso BEC simulato, su log di Microsoft 365 forniti dal docente:

- analisi dei log per individuare accessi anomali per provenienza geografica e orario;
- individuazione delle regole di inoltro e dei consensi OAuth malevoli;
- cronologia della compromissione, con esportazione verbalizzata dei log.

La parte B riguarda il rilevamento dei deepfake. I gruppi valutano con detector open source un insieme di immagini e registrazioni audio, autentiche e sintetiche, misurano falsi positivi e falsi negativi prima e dopo la ricompressione dei file e discutono come riportare l'esito in una relazione tecnica che dichiari i limiti degli strumenti.

Consegna: cronologia della compromissione con l'esportazione verbalizzata dei log (parte A); tabella degli errori dei detector prima e dopo la ricompressione e paragrafo di relazione sui limiti del rilevamento (parte B).

10.11 Letture

- R. Ayers, S. Brothers, W. Jansen, *NIST SP 800-101 Rev. 1, Guidelines on Mobile Device Forensics*, 2014.
- Microsoft, documentazione di Microsoft Purview Audit e dello Unified Audit Log.
- NIST, *NISTIR 8006, NIST Cloud Computing Forensic Science Challenges*, 2020.
- H. Farid, *Photo Forensics*, MIT Press, 2016.
- C2PA, *C2PA Technical Specification* (Content Credentials).
- R. Chesney, D. Citron, «Deep Fakes: A Looming Challenge for Privacy, Democracy, and National Security», *California Law Review*, 107, 2019.
- Reg. (UE) 2023/1543 e Dir. (UE) 2023/1544 (e-Evidence).

10.12 Domande di autovalutazione

1. Quali vincoli giuridici limitano l'acquisizione dei dati di uno smartphone, e quali si aggiungono quando l'estrazione sfrutta vulnerabilità del dispositivo? *Traccia.* Ogni accesso

ai dati richiede una base giuridica e un provvedimento dell'autorità e deve essere proporzionato agli scopi dell'indagine, data la quantità di dati personali del titolare e di terzi, con controllo preventivo di un giudice o di un'autorità indipendente salvo urgenza (C-548/21). Si applicano inoltre le misure di conservazione dei dati originali (L. 48/2008) e, quando l'estrazione altera il dispositivo, il regime degli accertamenti non ripetibili. Quando si sfruttano vulnerabilità, come per l'acquisizione fisica e spesso per quella completa del file system, si aggiungono l'alterazione dello stato del dispositivo e l'onere di documentare strumento e metodo, la cui affidabilità può essere contestata.

2. Quali log cloud documentano la compromissione di un account? *Traccia.* I log di accesso di Entra ID (provenienza, dispositivo, esito della MFA), lo Unified Audit Log di Microsoft 365 (regole di posta, accessi alle caselle, consensi OAuth), CloudTrail e Activity Log per le chiamate API e la creazione di chiavi. Vanno esportati con hash e verbale entro i periodi di conservazione.
3. Perché l'esito di un deepfake detector non costituisce da solo una prova? *Traccia.* Il detector fornisce una probabilità con tassi di errore non trascurabili, più alti su generatori nuovi e su contenuti ricompresi. Esiste inoltre il liar's dividend. La conclusione richiede la convergenza con metadati, provenienza, contesto e testimonianze, e la relazione dichiara i limiti dello strumento.
4. Che cosa attesta la provenienza C2PA e che cosa non può attestare? *Traccia.* Attesta, con una firma verificabile, il dispositivo o il software d'origine e le modifiche dichiarate, e ne prova l'integrità. Non attesta la veridicità della scena, e la sua affidabilità dipende dal firmatario. La sua assenza non prova la falsità, perché il manifesto può mancare o essere stato rimosso.
5. Come cambia la catena di custodia per evidenze acquisite da cloud? *Traccia.* Al posto del supporto si custodisce l'esportazione: vanno documentati account usato, filtri, intervallo temporale, hash del file esportato e prova della completezza. Il fornitore è un soggetto terzo, la conservazione è limitata e la tempestività diventa parte della custodia; per i dati all'estero servono gli strumenti di cooperazione giudiziaria.

Capitolo 11. Malware analysis con AI

L'analisi del malware serve alla risposta agli incidenti quando occorre sapere che cosa fa un campione, come si diffonde, con quali server comunica e quali tracce lascia. Il capitolo descrive i tre livelli dell'analisi (statico, dinamico e reverse engineering) e, per ciascuno, il contributo e i limiti degli strumenti basati sull'intelligenza artificiale: classificatori di machine learning, sandbox con analisi automatica dei comportamenti, modelli linguistici affiancati al decompilatore. Gli approfondimenti trattano il packing, l'evasione delle sandbox, l'uso dei modelli linguistici nel reverse engineering, le regole YARA, il raggruppamento dei campioni in famiglie e l'organizzazione economica del ransomware. L'obiettivo formativo è la capacità di leggere, discutere e usare un report di malware analysis nel contesto di un incidente; la specializzazione nel reverse engineering richiede percorsi dedicati.

Percorso. Specializzazione Digital Forensics (capitoli 8–11). Letture di raccordo: § 3.9, 4.8 e 6.6.

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- condurre un'analisi statica e dinamica di base di un campione in ambiente isolato;
- impiegare funzioni di decompilazione e annotazione assistite dall'AI, valutandone l'affidabilità;
- interpretare i verdeti dei motori di classificazione basati su machine learning e collegarli alle azioni di contenimento.

11.1 Feature statiche e loro fragilità

L'analisi statica esamina il file senza eseguirlo. Per gli eseguibili Windows il riferimento è il formato **PE** (Portable Executable), per i sistemi Linux il formato ELF. Le informazioni ricavabili sono numerose: intestazioni e marcatura temporale di compilazione, sezioni (codice, dati, risorse) con le relative dimensioni, tabella delle funzioni importate dalle librerie di sistema, stringhe leggibili, risorse incorporate, firme digitali, indicatori della presenza di un packer.

Gli stessi elementi sono le caratteristiche (feature) su cui lavorano i classificatori di machine learning integrati nei prodotti antivirus ed EDR. Alcuni modelli usano feature progettate dagli analisti, come nel dataset pubblico EMBER (Anderson e Roth, 2018), che descrive ogni file PE con statistiche su intestazioni, sezioni, import e distribuzione dei byte. Altri apprendono direttamente dalla sequenza di byte del file, come l'architettura MalConv proposta da Raff e colleghi nel 2017. In entrambi i casi il vantaggio è la velocità: con verdeti in millisecondi si esegue il triage di grandi volumi di campioni.

Il **packing** altera le feature statiche: un eseguibile compresso o cifrato mostra all'analisi le caratteristiche dell'involucro e non del programma (§ 11.4). Con il concept drift le famiglie di malware cambiano più rapidamente dei dati di addestramento, e un modello valutato su campioni contemporanei a quelli di addestramento sovrastima la propria efficacia futura, come hanno mostrato gli studi sul bias temporale nella valutazione dei classificatori (Pendlebury e colleghi,

2019). Resta l'evasione intenzionale: un attaccante che conosce o interroga il modello può modificare il file senza alterarne il comportamento fino a ottenere un verdetto benigno (capitolo 13).

Il verdetto di un classificatore statico è quindi un indizio da confermare con l'analisi dinamica e, se necessario, con il reverse engineering. L'indicatore statico più semplice da interpretare resta l'entropia delle sezioni, che segnala la probabile presenza di codice compresso o cifrato.

11.2 Sandbox e tecniche di evasione

L'analisi dinamica esegue il campione in un ambiente controllato e ne osserva il comportamento. Una **sandbox** è una macchina virtuale strumentata che registra le chiamate di sistema, i processi creati, le modifiche al file system e al registro, i meccanismi di persistenza e il traffico di rete, compresi i tentativi di contattare server di comando e controllo (C2). CAPEv2, progetto open source derivato da Cuckoo Sandbox, aggiunge l'estrazione automatica dei payload e delle configurazioni per molte famiglie note; servizi commerciali come ANY.RUN offrono un'analisi interattiva, in cui l'analista può intervenire durante l'esecuzione. Si tratta di servizi cloud: l'invio di un campione raccolto durante un'indagine lo fa uscire dal perimetro e, senza la modalità privata prevista dai piani a pagamento, può rendere l'analisi consultabile da terzi, compreso l'attaccante.

Il report di una sandbox contiene comportamenti osservati e, spesso, un punteggio di pericolosità e la mappatura sulle tecniche di MITRE ATT&CK. Su questi dati operano modelli di machine learning che raggruppano i campioni per somiglianza di comportamento e li attribuiscono a famiglie. Il vantaggio rispetto alle feature statiche è che il comportamento è più difficile da mascherare del contenuto del file: un programma deve comunque cifrare i file, contattare il server o installare la persistenza per raggiungere il suo scopo.

Il limite principale è che il campione può accorgersi di essere osservato, e un report privo di comportamenti malevoli non ne dimostra quindi l'innocuità. Le tecniche di evasione, catalogate da ATT&CK come T1497 (Virtualization/Sandbox Evasion), comprendono il riconoscimento degli artefatti delle macchine virtuali, la misura dei tempi di esecuzione, l'attesa di un'interazione umana e l'attivazione condizionata a date o caratteristiche dell'ambiente; il § 11.5 le descrive in dettaglio. Le contromisure consistono in ambienti realistici, esecuzioni prolungate e ripetute con configurazioni diverse e nel confronto con l'analisi statica e della memoria.

11.3 L'LLM accanto al decompilatore

Il reverse engineering ricostruisce la logica di un programma a partire dal codice macchina. Il disassemblatore traduce i byte in istruzioni assembly; il decompilatore, come quello incluso in Ghidra (lo strumento pubblicato dalla NSA nel 2019) o in IDA Pro, produce pseudocodice in un linguaggio simile al C. Lo pseudocodice è in genere fedele alla logica del programma, senza garanzia di correttezza: errori nella ricostruzione di tipi, chiamate e flusso di controllo sono docu-

mentati anche nei decompilatori più diffusi. È inoltre difficile da leggere, per i nomi di funzione generati automaticamente, le variabili senza nome, le strutture dati non ricostruite e i costrutti introdotti dal compilatore.

I modelli linguistici si affiancano al decompilatore attraverso estensioni che inviano lo pseudo-codice di una funzione al modello e ne ricevono una spiegazione in linguaggio naturale, proposte di nomi per funzioni e variabili, l'identificazione di algoritmi noti e una sintesi del comportamento complessivo. Gli stessi strumenti aiutano a estrarre configurazioni e indicatori di compromissione (IoC) da usare nella detection.

Su codice offuscato o ambiguo il modello produce spiegazioni plausibili ma errate, inventa scopi per funzioni che non comprende e può attribuire a una cifratura personalizzata il nome di un algoritmo standard (§ 11.6). Il codice di un campione raccolto durante un'indagine è inoltre un dato d'indagine, e inviarlo a un servizio esterno lo fa uscire dal perimetro; per questo il laboratorio usa un modello eseguito in locale (capitolo 6). Nella relazione tecnica ogni affermazione va verificata sul disassemblato o con l'esecuzione, e il modello non compare come fonte delle conclusioni.

11.4 Packing ed entropia nell'analisi statica

Un **packer** comprime o cifra il codice originale e lo racchiude in un piccolo programma (stub) che, all'avvio, lo ricostruisce in memoria e gli trasferisce il controllo. I packer hanno anche usi legittimi, come la riduzione delle dimensioni o la protezione del software commerciale; nel malware servono a ostacolare l'analisi statica e il rilevamento basato su firme. ATT&CK li classifica come sotto-tecnica T1027.002 (Software Packing).

L'indicatore più semplice è l'**entropia** di Shannon calcolata sui byte di ciascuna sezione. Su un alfabeto di 256 valori l'entropia massima è di 8 bit per byte: dati compressi o cifrati si avvicinano a questo valore, mentre codice macchina e testo hanno valori sensibilmente inferiori. L'analisi dell'entropia come strumento per individuare malware compresso o cifrato è stata descritta da Lyda e Hamrock nel 2007. Altri indizi concordanti sono una tabella degli import ridotta a poche funzioni, spesso quelle che servono a caricare dinamicamente le librerie (LoadLibrary e GetProcAddress), sezioni con nomi insoliti e una sproporzione tra dimensione su disco e dimensione in memoria.

Scenario didattico. Si consideri un campione con una sezione di codice molto ridotta e una sezione dati di 2 MB con entropia 7,9, un profilo tipico di loader. L'analisi statica non rivela stringhe significative. Dopo l'esecuzione in ambiente controllato, il dump della memoria del processo mostra il payload decompresso, con centinaia di stringhe leggibili, indirizzi di wallet e l'URL del server di comando e controllo, assenti nel file su disco.

Il codice effettivo diventa quindi osservabile durante l'esecuzione o estraendolo dalla memoria (capitolo 9). I classificatori basati su caratteristiche statiche risentono del packing: sui file compressi le loro decisioni dipendono in larga parte dalle caratteristiche del packer, si degradano

con packer assenti dai dati di addestramento e possono classificare come malevoli programmi legittimi compressi (Aghakhani e colleghi, 2020); gli stessi modelli sono inoltre il bersaglio degli attacchi di evasion del capitolo 13.

11.5 Sandbox ed evasione dell'analisi dinamica

Molti campioni verificano l'ambiente prima di eseguire le funzioni malevole. Le verifiche più comuni riguardano:

- **artefatti di virtualizzazione:** driver, processi e chiavi di registro tipici degli hypervisor, prefissi degli indirizzi MAC assegnati ai fornitori di virtualizzazione, dimensioni ridotte di disco e memoria;
- **tempi:** un ambiente strumentato è più lento, e alcune istruzioni rivelano la virtualizzazione attraverso la loro durata;
- **segni di attività umana:** movimenti del mouse, documenti aperti di recente, cronologia del browser, tempo di attività del sistema;
- **condizioni esterne:** date, lingua e configurazione regionale del sistema, raggiungibilità di un dominio, presenza di un software specifico.

In alternativa il campione può semplicemente attendere più a lungo della finestra di osservazione della sandbox, che di norma dura pochi minuti.

Scenario didattico. Un dropper non mostra alcuna attività in tre esecuzioni in sandbox. Alla quarta, con orologio virtuale accelerato e un profilo utente realistico che comprende documenti, cronologia del browser e quaranta minuti di attività, scarica il payload. La differenza di comportamento a parità di campione indica una probabile evasione, da confermare individuando nel codice le verifiche sull'ambiente, ed è essa stessa un elemento da riportare nella relazione.

Prima di concludere per l'innocuità occorrono più esecuzioni con ambienti, durate e stimoli diversi, il confronto con l'analisi statica e con quella della memoria e la valutazione esplicita della possibilità di evasione. I modelli che riassumono automaticamente i report di sandbox ereditano lo stesso limite, perché sintetizzano solo i comportamenti che il campione ha mostrato.

11.6 Modelli linguistici nel reverse engineering

Il decompilatore di Ghidra assegna alle funzioni nomi composti dal prefisso FUN_ seguito dall'indirizzo e alle variabili nomi generici. Su questo materiale un modello linguistico può spiegare in linguaggio naturale che cosa fa una funzione e proporre nomi significativi per funzioni e variabili, da applicare nel progetto di analisi. Può anche riconoscere algoritmi noti dalla loro struttura, come RC4, Base64, implementazioni di AES e funzioni di hash, e riassumere blocchi di logica e il comportamento di insieme del campione.

Scenario didattico. Una funzione di 120 righe contiene un ciclo con operazioni XOR e una tabella di 256 byte inizializzata con una permutazione. Il modello la riconosce come RC4 e suggerisce che il buffer passato in ingresso sia la chiave. L'analista verifica l'ipotesi nel debugger, estrae la chiave, decifra la configurazione del campione e ricava i domini di comando e controllo.

Il risparmio di tempo riguarda il lavoro meccanico di lettura e annotazione. La ricostruzione dell'architettura del campione, la valutazione delle scelte dell'autore e l'individuazione dei punti in cui il codice è stato reso deliberatamente ingannevole restano compiti dell'analista. Gli errori tipici del modello sono il riconoscimento di schemi inesistenti, l'attribuzione di scopi plausibili a funzioni ambigue e la descrizione sicura di codice offuscato che non ha compreso. Per questo ogni ipotesi del modello viene trattata come tale e verificata sul disassemblato o con l'esecuzione controllata.

11.7 Regole YARA per descrivere una famiglia

YARA, sviluppato presso VirusTotal, è il linguaggio con cui gli analisti descrivono le caratteristiche che identificano un malware. Una regola contiene stringhe testuali, sequenze esadecimali di byte con caratteri jolly per le parti variabili ed espressioni regolari, combinate in una condizione logica che può fare riferimento anche alla struttura del file (per esempio attraverso il modulo che interpreta il formato PE).

Per generalizzare, una regola descrive ciò che l'autore del malware difficilmente modifica tra una versione e l'altra, come una routine distintiva o il formato della configurazione, e lascia da parte i dettagli che cambiano a ogni compilazione, come hash, URL e nomi di file. La regola va verificata in due direzioni: sui campioni noti della famiglia, per misurarne la copertura, e su un ampio insieme di file legittimi, per misurarne i falsi positivi, che in produzione provocano quarantene e interruzioni di servizio.

Scenario didattico. L'analisi del § 11.6 ha messo in luce una routine RC4 con una tabella di inizializzazione non standard. La regola codifica 24 byte di quella routine con due caratteri jolly sulle parti che variano tra le build. Applicata all'archivio interno dei campioni, individua 17 file della stessa famiglia raccolti in due anni, tre dei quali non erano mai stati collegati tra loro.

Un modello linguistico produce bozze di regole a partire da un campione o da un report, ma tende a scegliere stringhe fragili o troppo comuni. La scelta degli elementi distintivi richiede la conoscenza della famiglia: le proposte del modello vengono selezionate dall'analista e la regola viene provata sui due insiemi di file prima del rilascio, come per le regole Sigma del capitolo 3.

11.8 Famiglie di malware e clustering

I campioni raramente sono indipendenti: derivano gli uni dagli altri, condividono librerie e generatori (builder) e talvolta codice sorgente diffuso pubblicamente. Il **clustering** li raggruppa in famiglie con misure di similarità a più livelli:

- similarità del file, calcolata con funzioni di fuzzy hashing come ssdeep o TLSH (capitolo 8);
- similarità tra le funzioni del codice disassemblato, misurata da strumenti di confronto binario come BinDiff o la funzione BSim di Ghidra;
- sovrapposizione dei comportamenti osservati in sandbox;
- riuso dell'infrastruttura di rete, come domini, indirizzi e certificati.

Riconosciuta la famiglia, si riutilizzano analisi già svolte, come capacità note, formato della configurazione, strumenti di estrazione esistenti e tecniche attese dall'operatore, e i tempi di conferma si riducono.

Scenario didattico. Il loader di un incidente ha valori di ssdeep bassi rispetto all'intero archivio, ma il confronto a livello di funzioni, eseguito sul codice estratto dalla memoria, rileva circa il 70% di codice in comune con una famiglia nota: il campione era stato ricompilato e compresso con un packer diverso, e per questo il fuzzy hash calcolato sul file non aveva segnalato somiglianze. L'estrattore di configurazione di quella famiglia funziona sul campione e restituisce i server di comando e controllo.

Il riuso del codice richiede cautela nell'interpretazione. Dopo la diffusione del codice sorgente del ransomware Babuk nel 2021 sono comparse varianti usate da gruppi privi di legami con gli autori originali, e anche le librerie comuni aumentano la similarità apparente. La famiglia è un dato tecnico misurabile; l'identità di chi la usa è un'inferenza, da formulare con il grado di confidenza descritto nel capitolo 4 a proposito dell'attribuzione.

11.9 Organizzazione economica del ransomware

Il ransomware attuale è organizzato con una divisione del lavoro. Gli sviluppatori offrono il malware e l'infrastruttura in abbonamento o in cambio di una percentuale dei riscatti (**ransomware as a service**, RaaS). Gli accessi iniziali alle reti sono venduti da intermediari specializzati (initial access broker), e l'intrusione è condotta dagli affiliati; altri soggetti gestiscono le negoziazioni e i siti su cui vengono pubblicati i dati sottratti.

Il modello prevalente è la **doppia estorsione**, diffusa a partire dal 2019: i dati vengono esfiltrati prima della cifratura, così che il ripristino da backup non elimini la minaccia di pubblicazione. Per l'analista le domande rilevanti diventano quali dati sono stati sottratti, da quando l'attaccante era presente nella rete e con quali tecniche ha operato. Le risposte provengono soprattutto dall'analisi di memoria, timeline e traffico di rete (capitolo 9) e orientano le decisioni legali e di notifica (capitolo 15).

Scenario didattico. La nota di riscatto identifica la famiglia, e i log di rete mostrano 60 GB trasferiti verso un servizio di archiviazione cinque giorni prima della cifratura. La timeline colloca l'accesso iniziale tre settimane prima, tramite una VPN priva di autenticazione a più fattori. Il quadro per i decisori comprende i dati esfiltrati, con i relativi obblighi di notifica (artt. 33–34 GDPR), la durata della permanenza e il vettore da chiudere.

In alcuni casi i file si recuperano senza pagare: implementazioni crittografiche difettose, chiavi ancora presenti nella memoria dei sistemi accesi (capitolo 7), strumenti di decifratura pubblicati dal progetto No More Ransom, avviato nel 2016 da Europol, dalla polizia olandese e da aziende di sicurezza. La verifica di queste possibilità è un compito di analisi. L'eventuale pagamento resta una decisione dei vertici dell'organizzazione, che richiede valutazioni legali (comprese eventuali sanzioni internazionali nei confronti dei destinatari) ed etiche.

11.10 Laboratorio: estrazione di IoC da un campione didattico disarmato

L'ambiente di lavoro è isolato e usa le distribuzioni REMnux (Linux) e FLARE VM (Windows), con campioni didattici disarmati predisposti dal docente. Nessun campione esce dal laboratorio.

Le attività si susseguono in quest'ordine:

1. analisi statica con Ghidra e un assistente basato su modello linguistico eseguito in locale: struttura del file, stringhe, entropia, funzioni principali;
2. esecuzione del campione in CAPEv2 e lettura del report comportamentale;
3. confronto tra le previsioni dell'analisi statica, i comportamenti osservati e le affermazioni dell'assistente, con l'individuazione degli errori del modello;
4. estrazione degli IoC e proposta di una regola di detection (YARA o Sigma).

Consegna: breve report del campione con comportamento osservato e famiglia ipotizzata, motivando il grado di confidenza; IoC estratti e regola proposta, con i risultati della verifica su file legittimi; elenco documentato degli errori commessi dall'assistente.

11.11 Letture

- M. Sikorski, A. Honig, *Practical Malware Analysis*, No Starch Press, 2012.
- D. Ucci, L. Aniello, R. Baldoni, «Survey of machine learning techniques for malware analysis», *Computers & Security*, 81, 2019.
- H. S. Anderson, P. Roth, «EMBER: An Open Dataset for Training Static PE Malware Machine Learning Models», arXiv, 2018.
- F. Pendlebury et al., «TESSERACT: Eliminating Experimental Bias in Malware Classification across Space and Time», *USENIX Security Symposium*, 2019.
- R. Lyda, J. Hamrock, «Using Entropy Analysis to Find Encrypted and Packed Malware», *IEEE Security & Privacy*, 5(2), 2007.

- H. Aghakhani et al., «When Malware is Packin' Heat: Limits of Machine Learning Classifiers Based on Static Analysis Features», NDSS Symposium, 2020.
- Documentazione di CAPEv2 e di YARA; MITRE ATT&CK, tecniche T1027 e T1497.

11.12 Domande di autovalutazione

1. Quali feature statiche sono più robuste all'offuscamento? *Traccia.* Sono più robuste le caratteristiche che l'autore non può modificare senza cambiare il funzionamento: struttura di routine distintive e formato della configurazione, osservabili sul codice estratto dalla memoria; le proprietà dello stub identificano soltanto il packer, usato anche da software legittimo. Stringhe, import e hash cambiano facilmente; per questo i classificatori statici vanno combinati con l'analisi dinamica e con il confronto a livello di funzioni.
2. Perché un verdetto ML va corroborato con l'analisi dinamica? *Traccia.* Il classificatore statico osserva il file e non il comportamento: è sensibile al packing, al concept drift e alle modifiche costruite per eludere il modello. L'analisi dinamica mostra le azioni effettive del campione, sulle quali si conferma o si smentisce il verdetto prima delle azioni di contenimento.
3. Quali tecniche di evasione inficiano una sandbox? *Traccia.* Il riconoscimento degli artefatti di virtualizzazione, la misura dei tempi, la ricerca di segni di attività umana, le condizioni esterne (date, lingua, raggiungibilità di domini) e l'attesa oltre la finestra di osservazione. Si contrastano con ambienti realistici ed esecuzioni ripetute e prolungate.
4. Quali errori tipici commette un LLM nello spiegare codice decompilato? *Traccia.* Riconosce schemi inesistenti, attribuisce a cifrature personalizzate il nome di algoritmi standard, inventa scopi plausibili per funzioni ambigue e descrive con sicurezza codice offuscato che non ha compreso. Ogni affermazione va verificata sul disassemblato o con l'esecuzione e il modello non è citato come fonte.
5. Come si passa dall'analisi del campione alle azioni di contenimento? *Traccia.* L'analisi produce IoC e conoscenza del comportamento (persistenza, server di comando e controllo, movimento laterale). Gli IoC alimentano blocchi e ricerche sugli altri sistemi, le regole individuano altri campioni della famiglia, e il comportamento indica quali sistemi isolare e quali credenziali ruotare.

Capitolo 12. AI offensiva e minacce emergenti

Il capitolo esamina come l'intelligenza artificiale generativa modifica le capacità di chi attacca, allo scopo di ricavarne requisiti per la difesa. La trattazione resta concettuale: descrive le minacce attraverso i loro schemi e le loro conseguenze, i segnali da riconoscere e i controlli che le contrastano, senza indicazioni operative, e le esercitazioni riguardano l'analisi delle minacce, con esclusione della produzione di contenuti offensivi. Le sezioni teoriche trattano l'ingegneria sociale generativa, i livelli di autonomia degli attacchi e le contromisure di processo e di identità. Gli approfondimenti riguardano lo spear phishing su larga scala, le frodi con voce e volto sintetici, la riduzione della barriera d'ingresso al crimine informatico, l'esposizione delle informazioni pubbliche, i segnali strutturali dei messaggi ostili e la lettura critica dei rapporti sulla minaccia.

Percorso. Specializzazione Sicurezza dell'AI (capitoli 6 e 12–14). Letture di raccordo: § 4.3 e 4.8.

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- analizzare l'impatto dell'AI generativa su phishing, vishing e frodi basate sull'identità;
- valutare su basi documentate la minaccia di attacchi automatizzati, semi-autonomi e autonomi;
- progettare requisiti difensivi: verifiche fuori banda, rilevamento comportamentale e di identità, resilienza dei processi aziendali.

12.1 Scala e personalizzazione dell'ingegneria sociale

L'ingegneria sociale sfrutta la fiducia delle persone per ottenere accessi, informazioni o pagamenti. Fino a pochi anni fa esisteva un compromesso tra qualità e scala: un messaggio personalizzato e convincente richiedeva tempo di ricerca e di scrittura, mentre le campagne di massa erano generiche e spesso riconoscibili per errori linguistici. L'AI generativa riduce questo compromesso. I modelli linguistici producono testi corretti in molte lingue e con il registro adatto; con i sistemi di sintesi vocale e video si imitano voce e aspetto di una persona a partire da registrazioni pubbliche.

L'uso dei modelli linguistici da parte di attori malevoli è documentato da fonti pubbliche. Euro-pol ha analizzato nel 2023 l'impatto dei grandi modelli linguistici sulle attività criminali e sulle forze di polizia. Nel febbraio 2024 Microsoft e OpenAI hanno descritto l'uso dei propri servizi da parte di gruppi collegati a Stati, impiegati soprattutto per ricerca, traduzione, scrittura di testi e supporto alla programmazione. Nel gennaio 2025 il Google Threat Intelligence Group ha pubblicato un'analisi analoga sull'uso improprio di Gemini, concludendo che i modelli hanno aumentato la produttività degli attori osservati senza fornire loro capacità sostanzialmente nuove.

Il caso più citato di frode basata su contenuti sintetici risale al 2024: un dipendente della sede di Hong Kong della società di ingegneria Arup ha autorizzato trasferimenti per circa 25 milioni di dollari dopo una videoconferenza in cui gli altri partecipanti erano generati artificialmente (§ 12.5). Per il modello di minaccia di un'organizzazione la conseguenza è un aggiornamento di scala e qualità degli attacchi; le tattiche e le tecniche dell'attacco restano quelle descritte da MITRE ATT&CK.

12.2 Livelli di autonomia degli attacchi

Per valutare la minaccia è utile distinguere tre livelli di autonomia degli attacchi:

- **automatizzato**: script e strumenti eseguono passi predefiniti su larga scala, mentre una persona decide obiettivi e sequenza; è la prassi consolidata da molti anni;
- **semi-autonomo**: agenti basati su modelli linguistici pianificano singoli passi e li adattano alle risposte dell'ambiente, con una supervisione umana residua; capacità di questo tipo sono state dimostrate in contesti di ricerca e segnalate in rapporti pubblici di threat intelligence;
- **autonomo**: l'intera catena d'attacco si svolge senza intervento umano; esistono dimostrazioni parziali, e lo scenario va considerato nella pianificazione della difesa.

Caso documentato. Nel novembre 2025 Anthropic ha descritto una campagna di spionaggio, attribuita con elevata confidenza a un gruppo sostenuto dallo Stato cinese, rivolta a circa trenta organizzazioni. Secondo la società, un agente basato sul modello Claude ha svolto la maggior parte delle operazioni tecniche (tra l'80 e il 90 per cento), mentre gli operatori umani intervenivano nelle decisioni principali; lo stesso rapporto segnala errori dell'agente, come credenziali dichiarate valide che non funzionavano. Alcuni ricercatori hanno contestato la stima del grado di autonomia e la mancanza di indicatori tecnici pubblicati: il caso si colloca al livello semi-autonomo e va letto con il metodo del § 12.9.

Lo stesso spettro esiste in ambito difensivo. L'AI Cyber Challenge promossa dalla DARPA tra il 2023 e il 2025 ha messo a confronto sistemi che individuano e correggono vulnerabilità nel software in larga parte automaticamente.

La conseguenza per chi difende riguarda i tempi. Se una parte dell'attacco procede a velocità di macchina, la risposta non può dipendere esclusivamente da passaggi manuali, e questo è il principale argomento a favore dell'automazione difensiva descritta nel capitolo 5, con i limiti discussi in quel capitolo (reversibilità delle azioni, rischio di errori su larga scala, sorveglianza umana). La generazione automatica di varianti di malware, con lo stesso comportamento e firme sempre diverse, riduce inoltre l'efficacia del rilevamento basato su firme (capitolo 11).

12.3 Contromisure di processo e di identità

Le contromisure efficaci contro l'ingegneria sociale generativa sono quelle che restano valide anche quando le persone non riconoscono il falso.

I controlli di **processo** sono i più robusti. La verifica fuori banda conferma le richieste ad alto rischio (pagamenti, modifiche delle coordinate bancarie, reset delle credenziali) su un canale indipendente da quello della richiesta, per esempio richiamando un numero presente nella rubrica aziendale. La doppia autorizzazione e i limiti di importo non derogabili per urgenza impediscono che una singola persona, ingannata, completi l'operazione, e parole d'ordine concordate in anticipo proteggono le richieste vocali d'emergenza.

I controlli **tecnologici** spostano il rilevamento dal contenuto al comportamento e all'identità: anomalie negli accessi, dispositivi e posizioni insoliti, richieste che deviano dalle abitudini dell'utente. L'autenticazione a più fattori resistente al phishing, basata sugli standard FIDO2 e WebAuthn, lega la credenziale al dominio legittimo e la rende inutilizzabile su un sito contraffatto; contrasta inoltre l'invio ripetuto di richieste di approvazione (MFA fatigue), usato per esempio nell'intrusione ai danni di Uber del 2022.

Anche la **formazione** va aggiornata. L'indicazione di cercare errori di ortografia perde valore quando i testi fraudolenti sono corretti; la formazione efficace insegna le procedure di verifica e i segnali strutturali descritti nel § 12.8. Il caso dell'attacco a MGM Resorts del 2023, attribuito al gruppo Scattered Spider e iniziato con una telefonata all'help desk in cui gli attaccanti si presentavano come un dipendente, mostra il ruolo dei processi di verifica dell'identità per le richieste di supporto.

12.4 Spear phishing generativo su larga scala

Lo spear phishing è un messaggio fraudolento costruito su misura per un destinatario specifico. Prima dell'AI generativa richiedeva tempo di ricerca e di scrittura per ogni bersaglio ed era quindi riservato a obiettivi di valore. I modelli linguistici riducono questo costo: producono testi corretti, nella lingua del destinatario e con il registro adatto, a partire da informazioni pubbliche sull'organizzazione.

La personalizzazione diventa così possibile su larga scala, perché il costo marginale di un messaggio su misura si avvicina a quello di un messaggio generico. Scompare inoltre l'indizio degli errori linguistici, su cui per anni si è basata la formazione degli utenti.

Caso documentato. Uno studio sperimentale di Heiding e colleghi, diffuso come preprint nel 2024 e pubblicato nel 2026, ha confrontato messaggi di spear phishing generati e personalizzati automaticamente, messaggi scritti da esperti e messaggi generici di controllo, inviati ai 101 partecipanti dello studio. I messaggi automatici hanno ottenuto un tasso di clic del 54%, pari a quello dei messaggi scritti da esperti e nettamente superiore al 12% del gruppo di controllo.

Per la difesa il rilevamento si sposta dal testo al contesto: canale e infrastruttura di invio (dominio registrato di recente, esito dei controlli SPF, DKIM e DMARC), coerenza della richiesta con i processi aziendali e verifica su un canale indipendente. Anche i sistemi di filtraggio della posta impiegano modelli di classificazione del testo, con il limite che un testo ben scritto offre loro pochi indizi.

12.5 Voce e volto sintetici nelle frodi

Con la sintesi vocale si imita la voce di una persona a partire da registrazioni pubbliche di breve durata, come un intervento a un convegno o un video aziendale (per il sistema Voice Engine, OpenAI ha indicato nel 2024 un campione di 15 secondi), e i sistemi in tempo reale sostengono anche una conversazione. Le tecniche di sostituzione del volto (deepfake) applicano lo stesso principio al video. Il primo caso ampiamente riportato di frode con voce sintetica risale al 2019: secondo quanto riferito dall'assicuratore Euler Hermes al Wall Street Journal, l'amministratore delegato della controllata britannica di un gruppo energetico trasferì circa 220.000 euro credendo di eseguire un ordine del vertice della capogruppo tedesca (§ 10.6).

Caso documentato. Nel 2024 un dipendente della sede di Hong Kong della società Arup ha autorizzato trasferimenti per circa 25 milioni di dollari dopo una videoconferenza in cui gli altri partecipanti, compreso il direttore finanziario, erano generati artificialmente. La polizia di Hong Kong ha descritto il caso nel febbraio 2024; la società ne ha confermato il coinvolgimento nel maggio successivo.

L'attacco sfrutta la fiducia nel canale di comunicazione e nell'aspetto o nella voce dell'interlocutore, per cui il riconoscimento visivo e vocale cessa di essere una forma di verifica dell'identità.

Scenario didattico. Un operatore dell'help desk riceve la telefonata di un dirigente che dichiara di essere in viaggio e chiede il reset dell'autenticazione a più fattori. La voce è simile a quella reale, il numero chiamante è falsificato e la richiesta è urgente. La procedura prevede la richiamata al numero presente in rubrica e una domanda di verifica concordata; applicandola, l'operatore blocca il tentativo.

Le contromisure sono quelle di processo descritte nel § 12.3, che restano efficaci anche quando l'imitazione è credibile. Il riconoscimento forense dei contenuti sintetici è trattato nel capitolo 10.

12.6 Riduzione della barriera d'ingresso

L'effetto dell'AI generativa più documentato sul crimine informatico è la diffusione di capacità di livello medio. Attori con competenze limitate ottengono testi di phishing, siti fraudolenti e codice di supporto che in passato richiedevano esperienza. Nel 2023 sono stati pubblicizzati su forum criminali modelli presentati come privi di filtri di sicurezza, come WormGPT e FraudGPT, e diversi rapporti hanno documentato tentativi di aggirare i filtri dei modelli pubblici.

Sul piano tecnico, la generazione automatica di varianti di codice con lo stesso comportamento e firme sempre diverse riduce ulteriormente l'efficacia del rilevamento basato su firme e aumenta il peso del rilevamento comportamentale (capitoli 3 e 9).

Scenario didattico. Un gruppo con competenze modeste prepara in pochi giorni una campagna con testi in più lingue e componenti rigenerati a ogni invio. Le tecniche sono note da anni; cambiano i tempi di preparazione e quindi il numero di campagne che lo stesso gruppo può condurre in parallelo.

Nella valutazione va considerato che i rapporti commerciali tendono a enfatizzare il «malware basato sull'AI», mentre le evidenze pubbliche indicano soprattutto l'accelerazione di tecniche già catalogate in MITRE ATT&CK e l'aumento di scala, con conseguenze dirette sui tempi della risposta.

12.7 Informazioni pubbliche e ricognizione

Ogni attacco mirato comincia con la raccolta di informazioni sul bersaglio. Quando questa attività richiedeva tempo di analisi, era riservata agli obiettivi di maggior valore; l'automazione ne riduce il costo. Chi difende deve quindi considerare già note all'attaccante le informazioni che l'organizzazione rende pubbliche, come organigrammi, fornitori, tecnologie dichiarate e annunci di lavoro che descrivono l'infrastruttura.

La valutazione del rischio si concentra sui processi che devono restare sicuri anche quando l'attaccante conosce i dettagli interni, e sulla riduzione delle informazioni pubblicate senza necessità. Una richiesta di pagamento fraudolenta può citare correttamente il nome di chi autorizza i bonifici, e il controllo efficace protegge il processo con la verifica su canale indipendente e la doppia autorizzazione, senza presupporre che l'attaccante ignori questa informazione.

Le misure pratiche comprendono la revisione dei dettagli di ruolo nelle firme e-mail, delle tecnologie citate negli annunci di lavoro e degli organigrammi pubblicati. Chi ricopre ruoli sensibili (amministrazione, tesoreria, help desk, assistenti dei vertici) riceve una formazione mirata, e l'esposizione pubblica dell'organizzazione si verifica periodicamente nell'ambito dell'analisi delle minacce (capitolo 4).

12.8 Segnali strutturali dei messaggi ostili

Quando la qualità del testo non distingue più un messaggio legittimo da uno fraudolento, il rilevamento si basa sugli elementi che la generazione automatica non modifica. I segnali utili riguardano:

- **canale e infrastruttura:** dominio del mittente registrato di recente o simile ma non identico a quello atteso, esito negativo dei controlli di autenticazione della posta (SPF, DKIM, DMARC), indirizzo del mittente diverso dal nome visualizzato, numero chiamante non verificabile;
- **coerenza con i processi:** richieste che saltano i passaggi previsti, come una modifica delle coordinate bancarie comunicata via e-mail, o che chiedono di evitare un controllo in via eccezionale;
- **contesto:** coerenza della richiesta con le abitudini del mittente, con il canale usato e con il momento, e pressione sui tempi che scoraggia la verifica.

Scenario didattico. Un'e-mail priva di errori, apparentemente inviata dall'amministratore delegato, chiede un bonifico urgente (frode di tipo BEC). Il dominio è simile ma non identico a quello aziendale, la richiesta non segue il processo di pagamento e insiste sui tempi. Come prevede la regola aziendale per ogni pagamento, il dipendente formato verifica la richiesta sul canale concordato.

I gateway di posta valutano gli stessi segnali strutturali, compresi reputazione e comportamento abituale del mittente, e impiegano a loro volta modelli di classificazione. Questi filtri si affiancano alle segnalazioni degli utenti e alla verifica su canale indipendente, che resta il controllo principale.

12.9 Lettura critica dei rapporti sulla minaccia

Le descrizioni di un malware «autonomo e inarrestabile» basato sull'AI non trovano finora riscontro nelle evidenze pubbliche. I dati disponibili, compresi i rapporti citati nel § 12.1, indicano che l'AI generativa accelera attività già esistenti, come la preparazione di esche, l'allestimento di infrastrutture e la generazione di varianti, riduce la barriera d'ingresso e aumenta il numero di campagne.

Dal 2025 le fonti pubbliche documentano anche usi più avanzati. Il CERT ucraino (CERT-UA) e il Google Threat Intelligence Group hanno descritto malware che interrogano un modello linguistico durante l'esecuzione per generare comandi o riscrivere parti del proprio codice. Uno di essi, attribuito da CERT-UA con confidenza media al gruppo APT28, è stato impiegato contro obiettivi ucraini, e secondo Google un secondo è stato osservato in attività reali, mentre altri erano sperimentali o in fase di sviluppo. La dipendenza da un servizio esterno offre a sua volta punti di rilevamento, come il traffico verso le interfacce del modello. Nel maggio 2026 il Google Threat Intelligence Group ha riferito di ritenere con elevata confidenza che attori criminali avessero usato un modello per individuare una vulnerabilità zero-day e svilupparne l'exploit, in vista di una campagna di sfruttamento su larga scala; la divulgazione coordinata con il fornitore potrebbe averla impedita. Il caso descritto nel § 12.2 mostra inoltre il valore della verifica indipendente: la stima del grado di autonomia proviene dalla società che ha osservato la campagna ed è stata contestata da alcuni ricercatori.

Le tattiche e le tecniche descritte da MITRE ATT&CK restano valide e i controlli di base mantengono la loro efficacia: autenticazione a più fattori resistente al phishing, aggiornamenti tempestivi, segmentazione della rete, backup verificati, verifica su canale indipendente. L'adattamento richiesto alla difesa riguarda soprattutto velocità e volume, e motiva l'automazione trattata nei capitoli 5 e 6.

Scenario didattico. Si consideri un rapporto commerciale che descrive un «malware che si riscrive da solo grazie all'AI». I dettagli tecnici mostrano un loader che rigenera l'offuscamento a ogni build, tecnica nota da anni. La conseguenza per la difesa è un'ulteriore perdita di efficacia delle firme e una maggiore importanza del rilevamento comportamentale.

Per valutare un rapporto sulla minaccia si applica il metodo di valutazione delle fonti del capitolo 4: si separano le osservazioni tecniche dalle interpretazioni, si cercano conferme indipendenti, si considerano gli interessi commerciali di chi pubblica e si traducono le conclusioni in priorità operative, cioè quali controlli rafforzare, quali rilevamenti aggiungere, quali processi rivedere.

12.10 Laboratorio: analisi difensiva di messaggi e procedure

Il laboratorio è un'esercitazione di analisi difensiva e non prevede la produzione di contenuti offensivi. Il docente fornisce un corpus di messaggi di phishing tradizionali e di messaggi generati, già raccolti e resi innocui.

Gli studenti svolgono, nell'ordine, queste attività:

1. valutazione comparata dei messaggi del corpus;
2. individuazione dei segnali residui: incoerenze di contesto, di identità e di canale;
3. progettazione di una procedura di verifica fuori banda per le richieste di pagamento ad alto rischio;
4. prova della procedura contro gli scenari di voce e video sintetici discussi nel capitolo.

Consegna: griglia dei segnali residui osservati; procedura di verifica fuori banda in una pagina; descrizione di uno scenario in cui la procedura fallisce, con le ragioni del fallimento. L'ultima consegna richiede di dichiarare i limiti della procedura progettata, come avviene in una relazione professionale.

12.11 Letture

- ENISA, *Threat Landscape*, edizioni annuali dal 2023, sezioni dedicate all'intelligenza artificiale.
- Europol Innovation Lab, *ChatGPT: The impact of Large Language Models on Law Enforcement*, 2023.
- Microsoft Threat Intelligence e OpenAI, *Staying ahead of threat actors in the age of AI*, febbraio 2024.
- Google Threat Intelligence Group, *Adversarial Misuse of Generative AI*, gennaio 2025; *GTIG AI Threat Tracker: Advances in Threat Actor Usage of AI Tools*, novembre 2025; *GTIG AI Threat Tracker: Adversaries Leverage AI for Vulnerability Exploitation, Augmented Operations, and Initial Access*, maggio 2026.
- Anthropic, *Disrupting the first reported AI-orchestrated cyber espionage campaign*, novembre 2025, da leggere insieme alle osservazioni critiche pubblicate dopo la diffusione.
- F. Heiding, S. Lermen, A. Kao, B. Schneier, A. Vishwanath, *Evaluating Large Language Models' Capability to Launch Fully Automated Spear Phishing Campaigns: Validated on Human Subjects*, preprint arXiv:2412.00586, 2024; versione rivista: F. Heiding et al., *Evaluating Large Language Models' Ability to Automate Spear Phishing*, *Expert Systems with Applications*, 314, 2026.

- CISA, *Implementing Phishing-Resistant MFA*, 2022.

12.12 Domande di autovalutazione

1. Perché i segnali linguistici non bastano più contro il phishing? *Traccia.* I modelli linguistici producono testi corretti e personalizzati a costo ridotto, quindi gli errori di lingua non distinguono più i messaggi fraudolenti. Il rilevamento deve basarsi su canale, infrastruttura, coerenza con i processi e contesto, e sulla verifica su canale indipendente.
2. Quali processi aziendali sono più esposti ai deepfake? *Traccia.* I processi in cui un'identità verificata solo a vista o a voce autorizza un'azione: pagamenti e modifiche delle coordinate bancarie, reset delle credenziali tramite help desk, richieste urgenti dei vertici. Vanno protetti con verifiche fuori banda, doppia autorizzazione e parole d'ordine concordate.
3. Cosa distingue un attacco automatizzato da uno autonomo? *Traccia.* Nell'attacco automatizzato gli strumenti eseguono passi predefiniti e una persona decide obiettivi e sequenza; in quello autonomo l'intera catena si svolge senza intervento umano. Tra i due si colloca il livello semi-autonomo, con agenti che pianificano e adattano singoli passi sotto supervisione.
4. Quali contromisure organizzative resistono all'ingegneria sociale generativa? *Traccia.* Quelle che non dipendono dal riconoscimento del falso: verifica fuori banda, doppia autorizzazione, limiti non derogabili, parole d'ordine concordate, autenticazione resistente al phishing (FIDO2), formazione sulle procedure e sui segnali strutturali.
5. Come incide l'AI offensiva sul modello di minaccia di un CSIRT? *Traccia.* Aumentano scala, qualità e velocità degli attacchi, mentre tattiche e tecniche restano le stesse. Il modello di minaccia va aggiornato considerando la personalizzazione di massa, le frodi con contenuti sintetici e la riduzione dei tempi, con conseguenze sull'automazione della risposta e sulla priorità dei controlli di processo.

Capitolo 13. Adversarial ML e sicurezza dei sistemi AI

Il capitolo studia i sistemi di intelligenza artificiale come superficie d'attacco. Le sezioni teoriche ordinano gli attacchi per fase del ciclo di vita e per obiettivo, descrivono i rischi specifici delle applicazioni basate su modelli linguistici e le difese architetturali corrispondenti. Gli approfondimenti trattano l'evasion, il poisoning e le backdoor, la prompt injection diretta e indiretta, l'estrazione del modello e l'inferenza sui dati di addestramento, la catena di fornitura di modelli e dataset e i riferimenti per una valutazione sistematica. La trattazione resta al livello dei meccanismi e delle contromisure; le prove pratiche si svolgono solo su sistemi didattici autorizzati.

Percorso. Specializzazione Sicurezza dell'AI (capitoli 6 e 12–14). Letture di raccordo: § 3.2, 11.1 e 11.4.

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- classificare un attacco a un sistema di machine learning o a un'applicazione basata su un modello linguistico secondo MITRE ATLAS e la OWASP Top 10 for LLM Applications;
- valutare la robustezza di un modello con strumenti di test, su sistemi autorizzati;
- progettare mitigazioni architetturali: validazione dell'input, isolamento degli strumenti, controlli sull'output.

13.1 Fasi e obiettivi degli attacchi

Chi attacca un sistema di machine learning può agire durante l'**addestramento**, sui dati o sul processo di apprendimento, oppure durante l'**inferenza**, sugli input sottoposti al modello già addestrato o osservandone le risposte. Il secondo asse di classificazione, dopo la fase, è l'obiettivo: integrità delle decisioni, riservatezza del modello e dei dati di addestramento, disponibilità del servizio. La tassonomia del NIST (AI 100-2 E2025) adotta questi assi, distingue i sistemi predittivi da quelli generativi e aggiunge, per questi ultimi, l'uso improprio delle capacità del modello.

Le principali classi di attacco si collocano così:

- **poisoning** (addestramento; integrità o disponibilità): campioni malevoli etichettati come benigni nel dataset;
- **backdoor** (addestramento; integrità): un segnale nascosto che disattiva il rilevamento;
- **evasion** (inferenza; integrità): un campione di malware modificato per eludere il classificatore;
- **membership inference** (inferenza; riservatezza): stabilire se un record faceva parte dei dati di addestramento;

- **model extraction** (inferenza; riservatezza): replicare il modello interrogandolo sistematicamente;
- **negazione del servizio tramite input** (inferenza; disponibilità): input costruiti per saturare o degradare il sistema.

Un terzo asse riguarda la conoscenza dell'attaccante: completa (white-box) o limitata alle risposte del modello (black-box). La segretezza del modello offre una protezione parziale, perché esempi avversari costruiti su un modello surrogato si trasferiscono spesso al modello bersaglio (trasferibilità). La catena di fornitura di modelli e dataset pre-addestrati attraversa tutte le fasi (§ 13.8).

13.2 Prompt injection e rischi delle applicazioni LLM

Un modello linguistico riceve in un unico contesto testuale le istruzioni dello sviluppatore, la richiesta dell'utente e i contenuti da elaborare, e non dispone di un meccanismo strutturale che separi le istruzioni dai dati. La **prompt injection** (il termine si è diffuso nel 2022) sfrutta questa caratteristica: un testo inserito nel contesto viene interpretato come istruzione.

La OWASP Top 10 for LLM Applications 2025 ordina i rischi di queste applicazioni; per un SOC contano in particolare i seguenti:

- **injection diretta** (LLM01): l'utente manipola il modello con il proprio input; il jailbreak, che mira ad aggirare le restrizioni di sicurezza del modello, ne è una forma;
- **injection indiretta** (LLM01): le istruzioni arrivano dai dati che il modello legge, come pagine web, e-mail, documenti e alert;
- **gestione impropria dell'output** (LLM05): l'output del modello viene eseguito o visualizzato senza validazione, per esempio come comando, query o codice HTML;
- **eccesso di autonomia** (LLM06, excessive agency): l'agente dispone di strumenti, permessi o margini di decisione oltre il necessario.

Un assistente che analizza e-mail di phishing elabora anche le istruzioni inserite dall'attaccante, e un agente di triage con strumenti di scrittura può essere indotto ad agire. Poiché i dati che un SOC analizza provengono da fonti potenzialmente ostili, l'injection indiretta è il rischio principale per gli assistenti e gli agenti descritti nei capitoli 6 e 14.

Per la prompt injection non esiste a oggi una soluzione completa, come riconosce la stessa OWASP; la mitigazione è architetturale e combina più controlli (§ 13.3).

13.3 Mitigazione per architettura

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

Sul piano dei **dati e dell'addestramento** contano la sanificazione e la provenienza dei dataset, l'addestramento robusto, la validazione continua delle prestazioni e il controllo dei modelli pre-addestrati (§ 13.8).

Nelle **applicazioni basate su modelli linguistici** istruzioni e dati vanno separati esplicitamente e ogni contenuto elaborato va trattato come ostile. Gli strumenti dell'agente ricevono il minimo privilegio, le azioni di scrittura o irreversibili richiedono l'approvazione umana e l'output si valida prima di ogni esecuzione o visualizzazione. Un criterio di progetto formulato nel 2025 (da Simon Willison e, in forma analoga, da Meta con la «Agents Rule of Two») considera tre capacità di un agente: elaborare contenuti non affidabili, accedere a dati riservati o a sistemi sensibili, comunicare verso l'esterno o modificare lo stato dei sistemi. In un agente che le riunisce tutte un'injection riuscita può tradursi in esfiltrazione o in azione; il progetto ne rimuove almeno una oppure introduce un'approvazione umana nel punto in cui si combinano.

La **verifica continua** comprende il red teaming periodico delle applicazioni con strumenti come Garak e PyRIT, i test di robustezza dei classificatori con Adversarial Robustness Toolbox e Counterfit e il monitoraggio delle interrogazioni anomale, che possono indicare tentativi di estrazione (§ 13.7).

13.4 Evasion ed esempi avversari

Un **esempio avversario** (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. Gli attacchi di evasione contro classificatori usati in sicurezza, come i filtri antispam, sono studiati dalla metà degli anni 2000 (Dalvi e colleghi, 2004; Lowd e Meek, 2005). Nel 2013 Biggio e colleghi hanno mostrato attacchi basati sul gradiente contro classificatori non lineari e, indipendentemente, nel 2014 Szegedy e colleghi hanno descritto il fenomeno per le reti neurali di visione artificiale. Nel 2015 Goodfellow, Shlens e Szegedy lo hanno collegato al comportamento quasi lineare dei modelli in spazi ad alta dimensionalità: vicino a un confine di decisione, perturbazioni piccole e mirate bastano ad attraversarlo.

Il fenomeno riguarda anche i modelli implementati correttamente, per cui un classificatore usato come controllo di sicurezza va misurato in condizioni avversarie (accuratezza robusta) oltre che sull'accuratezza media.

Nel dominio del malware l'attaccante deve rispettare vincoli assenti nelle immagini: il file modificato deve restare eseguibile e conservare il proprio comportamento (capitolo 11). La letteratura distingue perciò gli attacchi nello spazio delle caratteristiche da quelli nello spazio del problema, cioè sui file reali, più difficili da realizzare e più significativi per la valutazione.

Caso documentato. Nel 2019 i ricercatori di Skylight Cyber hanno mostrato che un antivirus commerciale basato su machine learning classificava come benigni file malevoli a cui erano stati aggiunti contenuti estratti da un programma legittimo, senza modificarne il comportamento. Il caso è incluso tra gli studi di caso di MITRE ATLAS.

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.

Tra le contromisure, l'addestramento con esempi avversari migliora la robustezza, con costi di calcolo e una possibile perdita di accuratezza. Vi si affiancano insiemi di modelli basati su caratteristiche indipendenti, per esempio statiche e comportamentali, e la limitazione del numero di interrogazioni e delle informazioni restituite. Nella difesa in profondità il classificatore è uno dei livelli di controllo, accanto al rilevamento comportamentale e alla sandbox.

13.5 Poisoning e backdoor nell'addestramento

A differenza dell'evasion, che colpisce un modello già addestrato, il **poisoning** interviene sull'apprendimento. Le prime analisi risalgono alla seconda metà degli anni 2000: Barreno e colleghi (2006) hanno proposto una tassonomia che distingue gli attacchi in fase di addestramento e Nelson e colleghi (2008) hanno mostrato il poisoning dei filtri antispam; nel 2012 Biggio, Nelson e Laskov hanno proposto un attacco di poisoning basato sul gradiente contro le macchine a vettori di supporto. Si distinguono le varianti seguenti:

- il poisoning **indiscriminato**, che contamina i dati per degradare le prestazioni complessive e colpisce quindi la disponibilità;
- il poisoning **mirato**, che altera la decisione su input specifici;
- la **backdoor**, in cui il modello si comporta normalmente salvo in presenza di un segnale scelto dall'attaccante (trigger); lo schema è stato descritto nel 2017 da Gu, Dolan-Gavitt e Garg con il lavoro BadNets.

Con i modelli addestrati su grandi raccolte di dati dal web il problema cambia scala. Carlini e colleghi (2024) hanno mostrato che dataset distribuiti come elenchi di indirizzi web possono essere contaminati da chi acquisisce il controllo di una parte delle fonti, e hanno proposto la verifica dell'integrità dei contenuti al momento del download. Nel 2024 Hubinger e colleghi hanno mostrato inoltre, con modelli linguistici in cui una backdoor era stata inserita deliberatamente («sleeper agents»), che il comportamento nascosto può sopravvivere al successivo addestramento di sicurezza.

Nel contesto di un SOC esiste una forma meno visibile, la manipolazione del ciclo di retroazione. Un sistema che riaddestra la propria linea di base sul comportamento osservato, come un sistema UEBA (capitolo 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.

Le linee di base vanno confrontate con riferimenti storici congelati oltre che con la finestra recente. Prima dell'addestramento si verificano la provenienza e l'integrità dei dati e si individuano i campioni anomali, mentre il modello addestrato si sottopone a test di regressione su insiemi di controllo fissi. Il monitoraggio del concept drift deve tenere conto che la deriva può essere provocata intenzionalmente (capitolo 2).

13.6 Prompt injection diretta e indiretta

La variante **diretta** della prompt injection proviene dall'utente, che cerca di far ignorare al modello le istruzioni dello sviluppatore o le restrizioni di sicurezza. Nella variante **indiretta** il testo rivolto al modello si trova nei dati: un'e-mail di phishing analizzata da un assistente, una pagina web riassunta, un documento recuperato da un archivio o un log contaminato. Greshake e colleghi ne hanno dato una descrizione sistematica nel 2023, con prove su applicazioni reali.

Il rischio cresce con le capacità dell'applicazione: un'injection riuscita altera il testo prodotto da un assistente che si limita a rispondere e può tradursi in azioni quando l'agente dispone di strumenti e permessi (capitolo 6).

Caso documentato. Nel giugno 2025 Aim Security ha descritto EchoLeak (CVE-2025-32711), una vulnerabilità di Microsoft 365 Copilot: un'e-mail con istruzioni nascoste, recuperata dall'assistente come contesto per rispondere a un utente, poteva indurlo a esporre dati interni senza che l'utente aprisse il messaggio. Microsoft ha corretto il servizio prima della divulgazione pubblica.

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.

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.

13.7 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; Tramèr e colleghi lo hanno dimostrato nel 2016 su servizi commerciali di machine learning accessibili tramite API.

Gli attacchi di **membership inference**, formalizzati da Shokri e colleghi nel 2017, stimano se un determinato record faceva parte dei dati di addestramento, con implicazioni di riservatezza quando i dati sono sanitari, giudiziari o comunque personali. Nei modelli linguistici il rischio si

estende ai contenuti memorizzati: nel 2021 Carlini e colleghi hanno recuperato da GPT-2 sequenze presenti alla lettera nei dati di addestramento, compresi dati personali.

Per la risposta agli incidenti un sistema AI esposto è anche una superficie da monitorare. I segnali di un'estrazione riguardano lo schema delle interrogazioni: volumi elevati da pochi account o indirizzi, copertura sistematica dello spazio degli input, input sintetici o poco plausibili per l'uso ordinario, richieste ripetute delle probabilità dettagliate, assenza dei ritmi tipici dell'uso umano.

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.

Le API dei modelli richiedono autenticazione e limitazione delle richieste, e le risposte contengono solo le informazioni necessarie, per esempio la classe senza le probabilità dettagliate. La privacy differenziale nell'addestramento (tecnica DP-SGD, formulata nel 2013 e applicata alle reti neurali profonde da Abadi e colleghi nel 2016) limita la memorizzazione dei singoli record. I modelli esposti entrano nel programma di monitoraggio della sicurezza. Con dati personali, la membership inference riguarda anche il GDPR (capitolo 15).

13.8 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 (OWASP LLM03 e LLM04).

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. Nel febbraio 2024 JFrog ha segnalato su Hugging Face modelli i cui file eseguivano codice al caricamento. Dalla versione 2.6 (gennaio 2025) PyTorch carica per impostazione predefinita solo i pesi; il formato safetensors, che contiene esclusivamente tensori, evita l'esecuzione di codice in fase di caricamento.

Una backdoor inserita nei pesi (§ 13.5) è difficile da individuare con un'ispezione. Dipendenze, tokenizer e file di configurazione allegati ampliano la superficie, così come le opzioni di alcune librerie che eseguono codice distribuito insieme al modello, per esempio `trust_remote_code` in Hugging Face Transformers, da abilitare solo dopo una revisione del codice.

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.

Si adottano formati di pesi non eseguibili come safetensors e si verificano provenienza e hash dei modelli. Il primo caricamento avviene in ambiente isolato, con l'analisi delle backdoor dove possibile. L'inventario dei modelli in uso (origine, versione, responsabile) ricalca la distinta dei componenti software (SBOM), che i formati CycloneDX e SPDX estendono a modelli e dataset.

13.9 Threat modeling con OWASP e MITRE ATLAS

La **OWASP Top 10 for LLM Applications** (versione 2025) elenca dieci rischi: prompt injection (LLM01), divulgazione di informazioni sensibili (LLM02), supply chain (LLM03), avvelenamento di dati e modelli (LLM04), gestione impropria dell'output (LLM05), eccesso di autonomia (LLM06), divulgazione del prompt di sistema (LLM07), debolezze di vettori ed embedding (LLM08), disinformazione (LLM09) e consumo illimitato di risorse (LLM10).

MITRE ATLAS (Adversarial Threat Landscape for Artificial-Intelligence Systems) deriva dalla Adversarial ML Threat Matrix pubblicata nel 2020 da MITRE e Microsoft con altre organizzazioni. Riprende la struttura di ATT&CK e ne riassume le tattiche dove applicabili, così che la copertura difensiva si mappa allo stesso modo. Vi aggiunge tattiche e tecniche specifiche dei sistemi di machine learning, come l'accesso al modello e la preparazione dell'attacco, l'evasion, il poisoning, l'estrazione e la prompt injection, insieme a mitigazioni e studi di caso su sistemi reali.

Il **threat modeling** del sistema AI usa questi riferimenti per individuare da dove entrano dati e istruzioni, quali componenti si fidano di quali altri, quali strumenti e permessi sono disponibili e dove un input ostile può propagarsi. Le tecniche tradizionali, come STRIDE e i diagrammi di flusso dei dati, si applicano aggiungendo agli elementi del modello i dataset, i pesi, i prompt di sistema, gli archivi consultati dal modello e gli strumenti dell'agente.

Scenario didattico. Prima della messa in esercizio dell'assistente SOC del capitolo 6, il threat modeling con OWASP e ATLAS rileva che l'agente legge e-mail non affidabili e può chiudere i ticket, un rischio LLM01 ad alto impatto che si mitiga con la separazione tra dati e istruzioni, l'approvazione umana per la chiusura e i test con Garak.

Nota normativa. Il regolamento europeo sull'intelligenza artificiale (Regolamento (UE) 2024/1689) richiede all'articolo 15, paragrafo 5, che i sistemi ad alto rischio siano resilienti ai tentativi di terzi non autorizzati di modificarne l'uso, gli output o le prestazioni sfruttandone le vulnerabilità, e cita espressamente l'avvelenamento dei dati e dei modelli, gli «esempi antagonisti» o «evasione dal modello», gli attacchi alla riservatezza e i difetti del modello. Il Regolamento (UE) 2026/1744 dell'8 luglio 2026 ha rinviato l'applicazione degli obblighi per i sistemi ad alto rischio, compreso l'articolo 15, al 2 dicembre 2027 per i sistemi dell'allegato III e al 2 agosto 2028 per quelli dell'allegato I. Il quadro normativo è trattato nel capitolo 15.

13.10 Laboratorio: red teaming controllato su sistemi didattici

Il lavoro pratico riguarda esclusivamente i sistemi predisposti dal docente: un classificatore didattico e un'applicazione basata su un modello linguistico con un insieme limitato di strumenti. Le prove su servizi di terzi senza autorizzazione sono escluse: oltre a violare i termini d'uso,

possono integrare reati come l'accesso abusivo a un sistema informatico (art. 615-ter del codice penale).

Gli studenti svolgono, nell'ordine, le attività seguenti:

1. valutazione di robustezza del classificatore con Adversarial Robustness Toolbox, confrontando accuratezza ordinaria e accuratezza robusta;
2. red teaming dell'applicazione didattica con Garak e PyRIT;
3. scenario guidato di prompt injection indiretta contro l'applicazione fornita dal docente;
4. scheda di threat model dell'applicazione: punti di ingresso, relazioni di fiducia tra componenti, tre rischi prioritari classificati secondo OWASP e ATLAS con la relativa mitigazione;
5. relazione dei risultati con le mitigazioni proposte, nel formato di un penetration test.

Consegna: risultati classificati secondo ATLAS e OWASP; scheda di threat model; mitigazione architetturale per ogni risultato; robustezza del classificatore prima e dopo le mitigazioni.

13.11 Letture

- MITRE, *ATLAS: Adversarial Threat Landscape for Artificial-Intelligence Systems*, atlas.mitre.org.
- OWASP, *Top 10 for LLM Applications 2025*.
- NIST, *AI 100-2 E2025, Adversarial Machine Learning: A Taxonomy and Terminology of Attacks and Mitigations*, 2025.
- B. Biggio, F. Roli, *Wild Patterns: Ten Years After the Rise of Adversarial Machine Learning*, Pattern Recognition, 2018.
- K. Greshake et al., *Not What You've Signed Up For: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection*, 2023.

13.12 Domande di autovalutazione

1. Che differenza c'è tra poisoning ed evasion? *Traccia.* Il poisoning agisce in fase di addestramento, sui dati o sul processo di apprendimento, e il suo effetto resta incorporato nel modello; l'evasion agisce in fase di inferenza su un modello già addestrato, modificando i singoli input. Contro il poisoning servono provenienza dei dati, riferimenti storici congelati e test di regressione, mentre l'evasion si contrasta con addestramento robusto, modelli complementari e difesa in profondità.
2. Perché la prompt injection indiretta è critica per gli agenti con tool? *Traccia.* Le istruzioni ostili arrivano dai dati elaborati dall'agente, che in un SOC provengono spesso da fonti ostili. Se l'agente dispone di strumenti con permessi di scrittura o di comunicazione verso l'esterno, un'injection riuscita si traduce in azioni o in esfiltrazione; la mitigazione limita privilegi e combinazioni di capacità e introduce l'approvazione umana.
3. Quali segnali indicano un'avvenuta model extraction? *Traccia.* Nei registri delle API compaiono gli schemi di interrogazione descritti nel § 13.7; all'esterno possono comparire servizi con un comportamento molto simile a quello del proprio modello.

4. Come si applica il principio del minimo privilegio a un agente LLM? *Traccia.* Si concedono solo gli strumenti necessari al compito, con permessi di sola lettura dove possibile e credenziali dedicate, e le azioni di scrittura o irreversibili richiedono approvazione umana. Si evita inoltre che lo stesso agente elabori contenuti non affidabili, acceda a dati riservati e comunichi verso l'esterno senza un punto di controllo.
5. Cosa aggiunge ATLAS rispetto ad ATT&CK? *Traccia.* Tattiche e tecniche specifiche dei sistemi di machine learning (accesso al modello, evasion, poisoning, estrazione, prompt injection), mitigazioni dedicate e studi di caso su sistemi reali, nella stessa struttura di ATT&CK.

Capitolo 14. Costruire l'AI difensiva e AI forensics

Il capitolo applica a un caso costruttivo i contenuti dei capitoli 6, 7 e 13. Descrive come si progetta un assistente di triage basato su Retrieval-Augmented Generation (RAG), come lo si valuta prima e dopo il rilascio e come lo si predispone all'indagine nel caso in cui sia coinvolto in un incidente. La registrazione degli eventi attraversa tutto il capitolo, perché un sistema AI che non conserva prompt, contesti, versioni e azioni non si può migliorare con metodo né analizzare a posteriori. Gli approfondimenti trattano gli aspetti tecnici che determinano la qualità del retrieval, i requisiti di osservabilità, i riferimenti normativi dell'AI Act e i criteri per scegliere tra sviluppo interno e acquisto.

Percorso. Specializzazione Sicurezza dell'AI (capitoli 6 e 12–14). Letture di raccordo: § 10.8.

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- creare un prototipo di assistente di triage con RAG su documentazione interna, eseguito con un modello locale;
- valutare un sistema di questo tipo prima del rilascio con metriche di groundedness, tasso di errore e latenza;
- analizzare un incidente che coinvolge un'applicazione basata su LLM a partire dal suo audit trail.

14.1 Assistente di triage basato su RAG

Un assistente per il SOC deve rispondere sulla base della documentazione dell'organizzazione: playbook e runbook, rapporti post-incidente, schede di threat intelligence, descrizioni dell'architettura. Un modello linguistico generico non conosce questi documenti e, se interrogato su di essi, tende a produrre risposte plausibili ma prive di fondamento. L'architettura RAG, proposta da Lewis e colleghi nel 2020, affianca al modello generativo un sistema di recupero dell'informazione; nel lavoro originale retriever e generatore erano sottoposti a fine-tuning congiunto, mentre nella forma oggi più diffusa, adottata in questo capitolo, i passaggi recuperati vengono inseriti nel prompt di un modello che non viene riaddestrato.

La catena di elaborazione procede per stadi:

- **corpus:** raccolta dei documenti da rendere interrogabili, con metadati su provenienza, data e livello di riservatezza;
- **indicizzazione:** suddivisione dei documenti in frammenti (chunk), calcolo di un embedding per ciascun frammento e memorizzazione in un database vettoriale;
- **retrieval:** a ogni domanda il sistema calcola l'embedding della richiesta e recupera i frammenti più simili, eventualmente combinando la ricerca vettoriale con quella per parole chiave;
- **generazione:** il modello riceve la domanda insieme ai frammenti recuperati e produce una risposta che li cita.

Per i dati d'indagine è preferibile un modello eseguito nell'infrastruttura dell'organizzazione, per esempio con Ollama in fase di prototipo o con un server di inferenza come vLLM in produzione, così che prompt e documenti non escano dal perimetro (capitolo 6).

Il confronto con il fine-tuning si basa sulla natura della conoscenza. Nel SOC le informazioni cambiano con frequenza e devono essere citabili: con il RAG l'aggiornamento richiede una nuova indicizzazione e ogni risposta indica le fonti. Il fine-tuning modifica i pesi del modello ed è adatto a fissare stile e formato dell'output più che ad aggiungere conoscenze, che resterebbero congelate alla data dell'addestramento. L'infrastruttura di inferenza è a sua volta un servizio critico: un server di modelli esposto senza autenticazione, un modello scaricato da una fonte non verificata (capitolo 13) o un consumo di risorse non limitato sono rischi da trattare come per ogni altro componente.

14.2 Valutazione prima del rilascio

Prima del rilascio il sistema va misurato su un insieme di test costruito dagli analisti. La **groundedness** indica la quota di affermazioni della risposta sostenute dai passaggi recuperati; si stima confrontando ogni affermazione con il contesto fornito al modello. Il **tasso di errore** si misura su casi storici anonimizzati con risposta nota, fissando prima dei test le soglie di accettazione, per evitare di adattarle ai risultati. La regressione riguarda il comportamento nel tempo: ogni modifica a prompt, modello o indice comporta la riesecuzione della suite, e per questo i prompt si gestiscono sotto controllo di versione come il codice. Contano anche la latenza, perché un assistente lento non viene usato durante un incidente, e il tasso di rifiuto corretto, cioè la capacità di dichiarare che la risposta non si trova nei documenti.

La valutazione prosegue dopo il rilascio: campionamento umano periodico degli output, monitoraggio della latenza e della deriva del corpus, canale di segnalazione per gli analisti che incontrano risposte errate. Il NIST AI Risk Management Framework (AI RMF 1.0, 2023) colloca il monitoraggio in produzione nella funzione Measure (MEASURE 2.4) e i piani di monitoraggio successivi al rilascio, con la raccolta del riscontro degli utenti, nella funzione Manage (MANAGE 4.1), e il relativo Playbook ne propone esempi operativi. Per i sistemi generativi il profilo NIST AI 600-1 (2024) individua rischi specifici, tra cui la produzione di contenuti non fondati.

14.3 Indagine sul sistema AI

Quando un sistema AI è coinvolto in un incidente occorre stabilire che cosa ha ricevuto, che cosa ha prodotto e quali azioni ha eseguito. La ricostruzione richiede che siano stati registrati:

- i prompt completi e i contesti recuperati dal RAG;
- le chiamate agli strumenti, con parametri ed esiti;
- le versioni del modello, del prompt di sistema e dell'indice;
- l'identità del richiedente e la marca temporale, con hash e regole di conservazione adeguate a un'evidenza.

L'indagine ricostruisce la catena causale che lega input, contesto, decisione e azione. Se si sospetta una prompt injection indiretta, si cerca nell'audit trail il documento che ha veicolato l'istruzione ostile e il momento in cui è entrato nell'indice. Per una decisione errata si accertano la versione in uso, il contesto fornito e l'output generato. I log del sistema AI sono evidenze digitali e si trattano con le regole del capitolo 7: acquisizione documentata, calcolo dell'hash, catena di custodia. Per i sistemi ad alto rischio l'art. 12 del Regolamento (UE) 2024/1689 (AI Act) richiede che il sistema consenta a livello tecnico la registrazione automatica degli eventi («log») per la durata del suo ciclo di vita; per i sistemi dell'allegato III l'obbligo si applica dal 2 dicembre 2027, a seguito del Regolamento (UE) 2026/1744.

14.4 Retrieval-Augmented Generation nel SOC

Il RAG risponde a limiti noti dei modelli linguistici, che non conoscono i dati dell'organizzazione e, quando l'informazione manca, generano risposte plausibili ma infondate. Recuperando i passaggi pertinenti e chiedendo al modello di rispondere solo sulla base di quelli, il sistema rende ogni risposta verificabile attraverso le citazioni.

Nel SOC conta anzitutto l'aggiornamento: un nuovo rapporto post-incidente diventa interrogabile dopo l'indicizzazione, senza addestramento. Le citazioni rendono la risposta tracciabile, perché l'analista apre i documenti citati e la verifica. Quanto al controllo degli accessi, se l'indice conserva i permessi dei documenti il retrieval li applica e ciascun utente ottiene risposte solo dai documenti che può consultare; un indice che ignora i permessi trasforma l'assistente in un canale di divulgazione di informazioni riservate.

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.

Se l'indice contiene documenti obsoleti, le risposte riportano indicazioni superate, pur con citazioni corrette. La base documentale è inoltre una superficie d'attacco, perché chi può scrivervi può inserire istruzioni ostili che il retrieval porta nel prompt (prompt injection indiretta, capitolo 13); la OWASP Top 10 for LLM Applications 2025 dedica una voce specifica alle debolezze dei vettori e degli embedding (LLM08).

14.5 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 almeno la fedeltà delle affermazioni ai passaggi recuperati, la copertura delle fonti pertinenti e il tasso di rifiuto corretto; framework open source come RAGAS forniscono implementazioni di misure di questo tipo, da tarare sul proprio dominio.

Il **red teaming** interno verifica la resistenza del sistema a prompt injection e jailbreak con strumenti come Garak (NVIDIA) e PyRIT (Microsoft), integrati da prove manuali sugli scenari più rilevanti per l'organizzazione. In produzione si raccoglie il riscontro degli analisti: ogni risposta errata segnalata e confermata in revisione diventa un nuovo caso del golden set, secondo lo stesso principio delle lezioni apprese (capitolo 1).

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.

Un modello linguistico usato come valutatore (LLM-as-judge) esamina molti output, ma la letteratura ha documentato distorsioni sistematiche, come la preferenza per le risposte più lunghe e la sensibilità all'ordine di presentazione (Zheng et al., 2023). Il valutatore automatico va quindi calibrato su un campione di giudizi umani, verificando periodicamente l'accordo tra valutazione automatica e valutazione degli analisti.

14.6 Analisi forense dei sistemi AI

L'analisi forense dei sistemi AI, un ambito in cui la letteratura specifica sui sistemi basati su LLM è ancora in formazione, si occupa anzitutto del sistema coinvolto in un incidente: una decisione errata che ha causato un danno, un agente manipolato tramite prompt injection, 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 al momento del fatto e il registro delle azioni dell'agente. L'altro oggetto dell'analisi sono le evidenze generate dall'AI: metadati e filigrane dei contenuti sintetici e credenziali di provenienza secondo lo standard C2PA (capitolo 10).

Vale il principio della forensic readiness (§ 10.8): i dati non registrati al momento dei fatti non possono essere acquisiti a posteriori. Per i sistemi AI l'acquisizione coincide in larga parte con la registrazione progettata in fase di sviluppo.

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 e delimita così le segnalazioni da rivedere; i log indicano versione del modello e orari. Il confronto tra versioni dell'indice individua il vettore d'ingresso, un documento alterato caricato il giorno prima da un account compromesso.

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.

14.7 Chunking, retrieval e reranking in un RAG

La qualità di un RAG dipende anzitutto dal **chunking**, che stabilisce come si suddividono i documenti. I frammenti devono conservare il contesto necessario a interpretarli senza diluire la pertinenza né occupare troppo spazio nel contesto del modello. Per documenti strutturati come i playbook la suddivisione per sezione logica dà in genere risultati migliori di quella per numero fisso di caratteri, e una parziale sovrapposizione tra frammenti adiacenti riduce la perdita di informazioni ai confini.

Il retrieval puramente vettoriale, basato sulla somiglianza di significato, individua con difficoltà le corrispondenze esatte, come un codice di errore, un nome di processo o un indirizzo IP. Per questo si usa spesso la ricerca ibrida, che combina la similarità tra embedding con un punteggio per parole chiave (per esempio BM25). Il **reranking** impiega infine un secondo modello, di solito un cross-encoder, 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 (accesso negato in Windows) 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.

Quando un RAG produce una risposta errata la diagnosi procede per stadi. 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 (§ 14.5).

14.8 Osservabilità e registrazione nei sistemi AI

L'osservabilità di un sistema AI si progetta prima del rilascio. Per ogni interazione si registrano:

- il prompt completo e la risposta, con hash e marca temporale;
- la versione del modello e del prompt di sistema in uso;
- i documenti recuperati dal RAG per quella risposta;
- le azioni dell'agente con il loro esito;
- il riscontro umano (accettazione, correzione, rifiuto).

La conservazione avviene su storage **append-only**, eventualmente con concatenazione degli hash tra record successivi, perché un log modificabile dopo l'incidente è esposto a contestazioni sull'integrità e ha un valore probatorio ridotto. Poiché i prompt contengono dati d'indagine e dati personali, il log è a sua volta un trattamento soggetto al GDPR (capitolo 15), con accessi tracciati, tempi di conservazione motivati e applicazione del principio di minimizzazione, che 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.

Oltre che al miglioramento del sistema e all'indagine post-incidente, la registrazione serve alla conformità: per i sistemi ad alto rischio l'AI Act richiede che il sistema consenta la registrazione automatica degli eventi (art. 12) e che fornitori (art. 19, par. 1) e deployer (art. 26, par. 6) conservino i log generati automaticamente, nella misura in cui sono sotto il loro controllo, per un periodo adeguato alla finalità prevista del sistema e di almeno sei mesi, salvo diversa disposizione del diritto dell'Unione o nazionale, in particolare in materia di protezione dei dati personali; per i sistemi dell'allegato III questi obblighi si applicano dal 2 dicembre 2027 (Regolamento (UE) 2026/1744).

14.9 Sviluppo interno o acquisto delle capacità AI

La maggior parte delle organizzazioni non addestra modelli propri e integra componenti esistenti; per ogni capacità sceglie tra sviluppo interno e acquisto. L'acquisto, per esempio la funzione AI di un SIEM o di un EDR o un servizio esterno, offre rapidità di adozione e manutenzione a carico del fornitore. I limiti riguardano la spiegabilità dell'output, il controllo sul comportamento del modello, la dipendenza dal fornitore e il possibile trasferimento di dati all'esterno. Lo sviluppo interno, per esempio un RAG su modello eseguito in locale come nel laboratorio, conserva il controllo dei dati e del comportamento al prezzo di costi di sviluppo, competenze e manutenzione.

In genere si acquistano le capacità mature e generiche, come il filtraggio dello spam o la detection di base sugli endpoint, e si sviluppano quelle che richiedono conoscenza aziendale e controllo del dato, come l'assistente sui propri playbook e rapporti post-incidente.

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.

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. In entrambi i casi restano necessari valutazione continua, osservabilità e un responsabile umano delle decisioni: l'organizzazione che usa il sistema mantiene gli obblighi propri del deployer e, se sviluppa il sistema e lo mette in servizio con il proprio nome anche solo per uso proprio, ne è anche fornitore ai sensi dell'AI Act (art. 3, punti 3 e 11; capitolo 15).

14.10 Laboratorio: prototipo RAG e incidente simulato

Nella parte A del laboratorio ogni gruppo costruisce un prototipo RAG su un corpus di playbook didattici, con un modello eseguito in locale, e configura una registrazione completa di prompt, contesti recuperati e versioni. Prepara poi una mini-suite di valutazione con dieci domande e risposta attesa e ne misura i risultati.

Nella parte B il docente inserisce nel corpus un documento contenente un'istruzione ostile (prompt injection indiretta). I gruppi, a partire dall'audit trail prodotto nella parte A, ricostruiscono il vettore d'ingresso, le risposte influenzate, i dati eventualmente esposti e le azioni indebite, e redigono una breve relazione forense. La parte B è realizzabile solo se nella parte A gli eventi sono stati registrati correttamente.

Consegna: prototipo funzionante con configurazione dei log; risultati della mini-suite di valutazione; relazione forense sull'incidente simulato.

14.11 Letture

- P. Lewis et al., *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*, NeurIPS 2020.
- NIST AI 100-1, *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*, 2023, e relativo *AI RMF Playbook*.
- NIST AI 600-1, *Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile*, 2024.
- OWASP, *Top 10 for LLM Applications*, edizione 2025.
- S. Es, J. James, L. Espinosa Anke, S. Schockaert, *RAGAs: Automated Evaluation of Retrieval Augmented Generation*, EACL 2024 (System Demonstrations), pp. 150–158; preprint arXiv:2309.15217, 2023.
- L. Zheng et al., *Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena*, NeurIPS 2023.
- Regolamento (UE) 2024/1689 (AI Act), artt. 12, 19 e 26, nel testo modificato dal Regolamento (UE) 2026/1744.

14.12 Domande di autovalutazione

1. Quando preferire il RAG al fine-tuning in un SOC? *Traccia.* Quando la conoscenza cambia spesso e le risposte devono essere verificabili: il RAG si aggiorna con una nuova indicizzazione, può citare le fonti e, se l'indice conserva i permessi, li applica al retrieval. Il fine-tuning è adatto a stile e formato dell'output e congela le conoscenze alla data dell'addestramento.
2. Quali log rendono indagabile un agente LLM? *Traccia.* Prompt completi e contesti recuperati, chiamate agli strumenti con parametri ed esiti, versioni di modello, prompt di sistema e indice, identità del richiedente e marca temporale, riscontro umano. I log vanno conservati su storage append-only con hash.
3. Come si definisce un test di regressione per i prompt? *Traccia.* Si costruisce un golden set di domande con risposta attesa, si fissano metriche e soglie prima dei test e si riesegue la suite

a ogni modifica di prompt, modello o indice; le risposte errate emerse in produzione diventano nuovi casi.

4. Quali rischi introduce l'infrastruttura di inferenza stessa? *Traccia.* Server di modelli esposti senza autenticazione, modelli e dipendenze di provenienza non verificata, consumo di risorse non limitato, log che contengono dati riservati e personali. L'infrastruttura va protetta e monitorata come ogni servizio critico.
5. Come si preserva l'integrità probatoria dei log di un sistema AI? *Traccia.* Con registrazione automatica su storage append-only, hash dei record ed eventuale concatenazione, sincronizzazione degli orologi, controllo e tracciamento degli accessi e acquisizione documentata secondo le regole della catena di custodia (capitolo 7).

Capitolo 15. Governance, etica e diritto

Il capitolo colloca l'impiego dell'intelligenza artificiale nella risposta agli incidenti all'interno del quadro normativo europeo e italiano e ne discute le implicazioni etiche. Gli strumenti presentati nel corso trattano dati personali, possono incidere sui diritti dei lavoratori e, se automatizzati, prendono decisioni con effetti concreti: il regolamento europeo sull'intelligenza artificiale, il GDPR, la disciplina NIS2 e lo Statuto dei lavoratori stabiliscono condizioni, obblighi e responsabilità. Il capitolo ha un taglio applicativo: per ogni norma indica la decisione di progetto che ne deriva. I riferimenti sono aggiornati alla data indicata nell'introduzione e vanno verificati nella versione vigente prima di ogni applicazione concreta.

Percorso. Traccia core, dopo la specializzazione. Letture di raccordo: § 5.1, 5.5, 5.6, 6.1, 6.5, 8.3, 8.6, 13.1, 13.9, 14.5 e 14.8, salvo quelle già studiate.

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- classificare un sistema AI di sicurezza secondo le categorie di rischio dell'AI Act;
- valutare la base giuridica del trattamento dei dati di telemetria, la necessità di una valutazione d'impatto e i limiti al controllo dei lavoratori;
- attribuire le responsabilità in caso di danno causato da un'azione automatizzata.

15.1 L'AI Act e gli strumenti di sicurezza

Il Regolamento (UE) 2024/1689 (AI Act) disciplina l'immissione sul mercato e l'uso dei sistemi di intelligenza artificiale secondo un approccio basato sul rischio. Distingue i ruoli del **fornitore**, che sviluppa un sistema o lo fa sviluppare e lo immette sul mercato o lo mette in servizio con il proprio nome o marchio, e del **deployer**, che lo utilizza sotto la propria autorità al di fuori di un'attività personale non professionale (art. 3). Un SOC che adotta uno strumento commerciale è di norma deployer; è anche fornitore se sviluppa un sistema e lo mette in servizio con il proprio nome, anche solo per uso proprio (art. 3, punto 11), e assume gli obblighi del fornitore nei casi dell'art. 25, per esempio con una modifica sostanziale di un sistema ad alto rischio.

Le categorie di rischio sono quattro:

- **pratiche vietate** (art. 5): tra queste la manipolazione con tecniche subliminali o ingannevoli, il social scoring e, di particolare interesse per il monitoraggio dei dipendenti, l'uso di sistemi di AI per inferire le emozioni di una persona sul luogo di lavoro e negli istituti di istruzione, salvo motivi medici o di sicurezza (art. 5, par. 1, lett. f); secondo gli orientamenti della Commissione del febbraio 2025 il divieto riguarda le inferenze basate su dati biometrici ed esclude l'analisi di testi scritti, e i motivi di sicurezza si riferiscono alla tutela della vita e della salute, con esclusione della protezione del patrimonio e della prevenzione delle frodi;
- **sistemi ad alto rischio** (art. 6): componenti di sicurezza di prodotti regolati (allegato I) e sistemi usati negli ambiti dell'allegato III, che comprende tra l'altro le infrastrutture critiche

e l'occupazione, compresi i sistemi destinati a monitorare e valutare il comportamento delle persone nei rapporti di lavoro;

- **obblighi di trasparenza** (art. 50) per alcuni sistemi, come quelli che interagiscono direttamente con le persone o generano contenuti sintetici;
- **rischio minimo**, per cui non sono previsti obblighi specifici oltre a quelli generali.

La maggior parte degli strumenti di detection e triage ricade nel rischio minimo. Il considerando 55 indica che i componenti destinati esclusivamente a fini di cibersicurezza non dovrebbero essere considerati componenti di sicurezza delle infrastrutture critiche ai fini dell'alto rischio (allegato III, punto 2). La classificazione cambia invece quando un sistema valuta il comportamento dei lavoratori o alimenta decisioni sul rapporto di lavoro. Per i sistemi ad alto rischio valgono i requisiti degli artt. 9–15 (gestione del rischio, governance dei dati, documentazione tecnica, registrazione degli eventi, trasparenza verso i deployer, sorveglianza umana, accuratezza, robustezza e cybersicurezza); l'art. 15 richiede in particolare misure contro avvelenamento dei dati, esempi avversari e attacchi alla riservatezza dei modelli, cioè le minacce trattate nel capitolo 13.

Il regolamento è entrato in vigore il 1° agosto 2024 e si applica per fasi: divieti e obbligo di alfabetizzazione dal 2 febbraio 2025, obblighi per i modelli di AI per finalità generali dal 2 agosto 2025, gran parte delle restanti disposizioni, compresi gli obblighi di trasparenza dell'art. 50, dal 2 agosto 2026. Il Regolamento (UE) 2026/1744 («Omnibus digitale sull'IA»), in vigore dal 27 luglio 2026, ha posticipato l'applicazione dei requisiti e degli obblighi per i sistemi ad alto rischio al 2 dicembre 2027 per quelli dell'allegato III e al 2 agosto 2028 per quelli legati ai prodotti dell'allegato I; per i sistemi di AI generativa immessi sul mercato prima del 2 agosto 2026 l'obbligo di marcatura dei contenuti previsto dall'art. 50, par. 2, si applica dal 2 dicembre 2026. Lo stesso regolamento ha riformulato l'obbligo di alfabetizzazione dell'art. 4 e aggiunto nuove pratiche vietate all'art. 5.

15.2 Obblighi del GDPR e della NIS2 per il SOC

I log di sicurezza contengono dati personali: indirizzi IP, identificativi degli account, orari di lavoro, contenuti di messaggi. Il loro trattamento richiede una base giuridica, di norma il legittimo interesse del titolare (art. 6, par. 1, lett. f, GDPR). Il considerando 49 riconosce come legittimo interesse il trattamento di dati relativi al traffico nella misura strettamente necessaria e proporzionata a garantire la sicurezza delle reti e dell'informazione, con riferimento ad autorità pubbliche, CERT, CSIRT e fornitori di reti, di servizi di comunicazione elettronica e di tecnologie e servizi di sicurezza. Resta necessario il bilanciamento con i diritti degli interessati. Valgono i principi dell'art. 5, in particolare minimizzazione e limitazione della conservazione.

Il monitoraggio esteso del comportamento degli utenti (UEBA) richiede quasi sempre una **valutazione d'impatto sulla protezione dei dati** (DPIA, art. 35): si tratta di un monitoraggio sistematico che riguarda persone in posizione di dipendenza, e l'elenco delle tipologie di trattamento soggette a DPIA adottato dal Garante (prov. n. 467 dell'11 ottobre 2018, allegato 1, punto 5) comprende i trattamenti che consentono il controllo a distanza dell'attività dei dipendenti. In

Italia si applica inoltre l'art. 4 della L. 300/1970 (Statuto dei lavoratori), esaminato nel § 15.8. L'invio di log o di testi a servizi di AI situati fuori dall'Unione è un trasferimento di dati soggetto alle regole del capo V del GDPR (artt. 44–49).

L'art. 22 tutela l'interessato dalle decisioni basate unicamente sul trattamento automatizzato che producono effetti giuridici o incidono in modo analogo significativamente sulla sua persona (§ 15.5). Sul versante della sicurezza, la Direttiva (UE) 2022/2555 (NIS2) attribuisce agli organi di amministrazione e direttivi dei soggetti essenziali e importanti l'approvazione delle misure di gestione del rischio e la supervisione della loro attuazione, con possibile responsabilità per le violazioni (art. 20; in Italia art. 23 del D.Lgs. 138/2024).

Il danno causato da un'azione automatizzata errata resta a carico dell'organizzazione che impiega il sistema; i contratti con i fornitori ne regolano i rapporti interni e la possibilità di rivalsa. Sul piano europeo la nuova direttiva sulla responsabilità per danno da prodotti difettosi (Dir. (UE) 2024/2853) include espressamente il software tra i prodotti. La direttiva, da recepire entro il 9 dicembre 2026, si applica ai prodotti immessi sul mercato o messi in servizio dopo tale data e riconosce il risarcimento alle sole persone fisiche, con esclusione dei danni a beni usati esclusivamente per scopi professionali e della perdita di dati usati a fini professionali; il danno subito da un'impresa, come nello scenario del § 15.6, resta regolato dal contratto e dalla responsabilità civile generale. Con l'audit trail descritto nel capitolo 14 si ricostruiscono e si motivano le decisioni.

15.3 Bias, proporzionalità, spiegabilità

Le norme fissano requisiti minimi; la valutazione etica di un sistema di sicurezza basato sull'AI considera anche altri aspetti. Un punteggio di rischio che penalizza sistematicamente un gruppo di persone (turnisti, un reparto, una sede) introduce un **bias** con effetti discriminatori, anche quando l'accuratezza media è elevata. Il bias si accerta con prove specifiche, per esempio confrontando tassi di falsi positivi e di segnalazione tra gruppi.

Il monitoraggio per finalità di sicurezza resta soggetto a necessità e **proporzionalità**, con raccolta dei dati minimi indispensabili, conservazione limitata e finalità dichiarate agli interessati. La spiegabilità è un requisito organizzativo e probatorio: ogni alert deve poter essere motivato verso l'analista che lo gestisce, verso la persona interessata e, se necessario, in giudizio (capitolo 8).

Questi temi sono presenti nei principali quadri di riferimento. Il quadro AI4People (Floridi et al., 2018) aggiunge ai principi della bioetica (beneficenza, non maleficenza, autonomia, giustizia) quello di esplicabilità. Il NIST AI Risk Management Framework (AI RMF 1.0, 2023) elenca tra le caratteristiche di un sistema affidabile la validità e affidabilità, la sicurezza per persone e beni (safe), la sicurezza informatica e la resilienza (secure and resilient), la responsabilità e trasparenza, la spiegabilità e interpretabilità, la tutela della privacy e l'equità con gestione dei bias dannosi.

Lo strumento che traduce questi principi in regole operative è la **policy d'uso dell'AI nel SOC**, redatta nel laboratorio (§ 15.10): inventario dei sistemi e loro classificazione, dati ammessi verso ciascun servizio, punti di revisione umana obbligatoria, requisiti di registrazione degli eventi e responsabilità nominative.

15.4 AI Act e strumenti di incident response

Per un SOC l'applicazione dell'AI Act parte da un censimento: ogni strumento con componenti AI va descritto, classificato e documentato. La classificazione non dipende dalla tecnologia ma dalla finalità e dal contesto d'uso. Un filtro antispam ricade di norma nel rischio minimo; un sistema che monitora e valuta il comportamento dei dipendenti, o che incide sulle decisioni relative al rapporto di lavoro, può rientrare nell'alto rischio (allegato III, punto 4). La deroga dell'art. 6, par. 3, per i sistemi che non presentano un rischio significativo non si applica quando il sistema effettua profilazione di persone fisiche, come accade di norma per un UEBA che attribuisce punteggi di rischio agli utenti. In questo caso registrazione degli eventi, documentazione e sorveglianza umana sono obblighi che si applicano dal 2 dicembre 2027 (Regolamento (UE) 2026/1744), e le sanzioni previste dall'art. 99 sono commisurate anche al fatturato mondiale annuo.

Scenario didattico. Si supponga che il censimento degli strumenti di un SOC individui quattordici sistemi con componenti AI. Dodici sono a rischio minimo o limitato; per l'UEBA che alimenta provvedimenti disciplinari e per il sistema antifrode interno si svolge l'analisi da alto rischio, con registro dei trattamenti aggiornato, log conservati e procedura di intervento umano scritta e verificata.

L'organizzazione che utilizza il sistema, anche quando lo acquista da un fornitore, è deployer e ha obblighi propri (art. 26): uso conforme alle istruzioni del fornitore, sorveglianza umana affidata a persone con competenza, formazione e autorità adeguate, monitoraggio del funzionamento, conservazione dei log generati automaticamente, informazione dei lavoratori e dei loro rappresentanti prima di mettere in servizio un sistema ad alto rischio sul luogo di lavoro.

15.5 Decisioni automatizzate e art. 22 GDPR

L'art. 22 GDPR riconosce all'interessato il diritto di non essere sottoposto a una decisione **basata unicamente sul trattamento automatizzato** che produca effetti giuridici o incida in modo analogo significativamente sulla sua persona. Secondo le linee guida del Gruppo di lavoro Articolo 29 (WP251 rev.01), fatte proprie dall'EDPB, e secondo la Corte di giustizia (sentenza del 7 dicembre 2023, causa C-634/21, SCHUFA) la disposizione stabilisce un divieto di principio, che si applica senza bisogno di un'iniziativa dell'interessato. Le eccezioni (decisione necessaria per un contratto, autorizzata dalla legge, basata sul consenso esplicito) richiedono misure adeguate a tutela dell'interessato: nei casi del contratto e del consenso esplicito almeno il diritto di ottenere l'intervento umano, di esprimere la propria opinione e di contestare la decisione (art. 22, par. 3); nel caso dell'autorizzazione per legge le misure sono stabilite dalla legge stessa (art. 22,

par. 2, lett. b). Le stesse linee guida precisano che l'intervento umano deve essere effettivo: chi rivede la decisione deve avere l'autorità e la competenza per modificarla e deve esaminare tutti i dati pertinenti.

Nella risposta agli incidenti un caso da valutare è la sospensione automatica dell'account di un dipendente segnalato dall'UEBA, con effetti sul rapporto di lavoro. Una ratifica sistematica delle proposte del sistema, condizionata dall'automation bias (capitolo 5), non costituisce un intervento umano effettivo.

Esempio. Nel playbook l'UEBA segnala, la piattaforma SOAR prepara la sospensione e l'esecuzione attende l'approvazione dell'analista, che esamina le evidenze e può dissentire; decisione e motivazione restano nel log. Il tempo di risposta resta nell'ordine dei minuti e la decisione non è basata unicamente sul trattamento automatizzato.

Nella risposta agli incidenti si applicano anche gli artt. 33 e 34: il titolare notifica la violazione all'autorità di controllo senza ingiustificato ritardo e, ove possibile, entro 72 ore dal momento in cui ne è venuto a conoscenza, salvo che sia improbabile che presenti un rischio per i diritti e le libertà delle persone, e la comunica agli interessati se il rischio è elevato (capitolo 1); il DPO supporta il titolare nella valutazione. L'AI può accelerare triage e stima dell'impatto; il testo notificato richiede comunque revisione umana. Nel novembre 2025 la Commissione ha proposto un pacchetto di modifiche («Digital Omnibus») che interviene anche sul GDPR, comprese le regole sulla notifica delle violazioni; a luglio 2026 la proposta è in negoziato e si applica il testo vigente.

15.6 Responsabilità per gli errori dei sistemi AI

Quando impiega un sistema automatico, l'organizzazione che lo utilizza resta responsabile insieme alle persone che ne decidono configurazione e uso. La catena delle responsabilità va definita in anticipo:

- il **fornitore** risponde del sistema come progettato, documentato e dichiarato;
- il **deployer** risponde dell'uso concreto, della configurazione, delle soglie e della scelta di automatizzare un'azione;
- il singolo operatore risponde delle decisioni che gli competono, nei limiti delle istruzioni ricevute.

La documentazione interna (chi ha autorizzato quale livello di autonomia, su quali valutazioni) deve esistere prima dei fatti, perché ricostruita a posteriori avrebbe scarso valore. Dopo un incidente la diligenza dell'organizzazione si valuta su questa documentazione e sul contratto con il fornitore (livelli di servizio, limiti dichiarati, gestione degli aggiornamenti).

Scenario didattico. Un contenimento automatico isola per errore i sistemi di cassa in un fine settimana di forte attività, con danno economico. L'istruttoria interna trova una matrice tra azioni e livelli di autonomia approvata dal CISO, una procedura di rollback verificata e un errore entro il tasso dichiarato dal fornitore. Il processo è quindi difendibile e il caso entra nel golden set (capitolo 14).

Oltre ai requisiti di legge restano questioni etiche: la distribuzione delle segnalazioni tra gruppi di persone, la proporzionalità del monitoraggio, la trasparenza verso chi è osservato.

15.7 NIS2 e obblighi di risposta agli incidenti

La Direttiva (UE) 2022/2555 (NIS2), recepita in Italia con il D.Lgs. 4 settembre 2024, n. 138, si applica ai soggetti essenziali e importanti dei settori elencati negli allegati, sotto la vigilanza dell'Agenzia per la cybersicurezza nazionale (ACN). Le misure di gestione del rischio richieste comprendono espressamente la gestione degli incidenti, la continuità operativa e la sicurezza della catena di approvvigionamento.

Per gli incidenti significativi è prevista una **notifica in più fasi** al CSIRT Italia (art. 25 del D.Lgs. 138/2024). La pre-notifica va trasmessa senza ingiustificato ritardo e comunque entro 24 ore da quando si è venuti a conoscenza dell'incidente. Entro 72 ore dalla stessa conoscenza segue la notifica, con una valutazione iniziale di gravità e impatto; una relazione intermedia può essere richiesta, e la relazione finale va trasmessa entro un mese dalla notifica. Gli organi di amministrazione e direttivi approvano le misure, ne sovrintendono l'attuazione, seguono una formazione specifica e possono essere chiamati a rispondere delle violazioni. Le sanzioni amministrative pecuniarie (art. 38 del D.Lgs. 138/2024, in attuazione dell'art. 34 della direttiva) possono raggiungere, per i soggetti essenziali, 10 milioni di euro o il 2% del fatturato annuo mondiale dell'esercizio precedente, se superiore, e per i soggetti importanti 7 milioni di euro o l'1,4%, se superiore; per le pubbliche amministrazioni il decreto prevede importi specifici.

Scenario didattico. Alle 22:00 il SOC viene a conoscenza di un incidente significativo: da quel momento decorre il termine di 24 ore per la pre-notifica. Con triage rapido e canale di notifica già predisposto il termine si rispetta, mentre un SOC che deve ancora stabilire destinatari e modalità rischia di superarlo. L'AI che accelera qualificazione e stima d'impatto (capitolo 5) contribuisce anche alla conformità.

Per rispettare i termini si predispongono in anticipo modelli di notifica, criteri scritti per stabilire la significatività, contatti dell'autorità nel playbook e ruoli assegnati. La notifica è un atto con rilievo giuridico e va preparata con la funzione legale. Uno stesso incidente può attivare insieme gli obblighi NIS2 e quelli degli artt. 33–34 GDPR, con destinatari e termini diversi.

15.8 Bias e monitoraggio dei lavoratori

Molti sistemi difensivi, come UEBA, prevenzione della perdita di dati (DLP) e rilevazione delle minacce interne, osservano comportamenti umani e sono quindi soggetti anche alle tutele delle persone.

Il **bias** nasce dai dati storici da cui apprende un modello che segnala anomalie: se alcuni ruoli, turni o sedi sono sovrarappresentati tra i casi segnalati in passato, il modello tende a perpetuare lo squilibrio. Oltre alle metriche aggregate di precision e recall (§ 2.4) occorre misurare i tassi di segnalazione e di falsi positivi per gruppo.

Quanto alla **proporzionalità**, in Italia il controllo a distanza dei lavoratori è disciplinato dall'art. 4 della L. 300/1970, come modificato dal D.Lgs. 14 settembre 2015, n. 151 e dal D.Lgs. 24 settembre 2016, n. 185. Gli strumenti da cui deriva anche la possibilità di controllo a distanza possono essere impiegati solo per esigenze organizzative e produttive, di sicurezza del lavoro e di tutela del patrimonio aziendale, previo accordo collettivo con le rappresentanze sindacali o, in mancanza di accordo, previa autorizzazione dell'Ispettorato nazionale del lavoro. La disciplina non si applica agli strumenti usati dal lavoratore per rendere la prestazione e a quelli di registrazione degli accessi e delle presenze. Le informazioni raccolte sono utilizzabili solo se il lavoratore ha ricevuto adeguata informazione e nel rispetto della normativa sulla protezione dei dati. La qualificazione degli strumenti di sicurezza informatica rispetto a queste ipotesi va valutata caso per caso, anche alla luce dei provvedimenti del Garante. La L. 23 settembre 2025, n. 132, in vigore dal 10 ottobre 2025, prevede inoltre che il datore di lavoro informi il lavoratore dell'utilizzo dell'intelligenza artificiale nei casi e con le modalità dell'art. 1-bis del D.Lgs. 26 maggio 1997, n. 152 (art. 11), e attribuisce all'Agenzia per l'Italia digitale e all'ACN il ruolo di autorità nazionali per l'intelligenza artificiale.

Scenario didattico. Un sistema per le minacce interne segnala con frequenza molto maggiore i dipendenti di un reparto. La verifica per gruppo mostra che il reparto lavora con orari irregolari per ragioni legittime e che tali orari sono sovrarappresentati tra i comportamenti anomali dei dati storici, ed evita così indagini disciplinari ingiustificate.

Le cautele operative comprendono informativa chiara ai lavoratori monitorati, anche sull'uso dell'AI (L. 132/2025, art. 11), accordo sindacale o autorizzazione dove richiesti, valutazione d'impatto per i trattamenti ad alto rischio, verifica periodica dei tassi di segnalazione per gruppo, con la regola che ogni segnalazione del sistema avvia una verifica umana senza costituire un accertamento.

15.9 Governance dell'AI nel SOC

Gli obblighi esaminati nel capitolo si traducono in una **governance** concreta, i cui elementi minimi sono:

- un inventario dei sistemi AI in uso con la relativa classificazione di rischio (§ 15.4);
- una policy d'uso che stabilisca quali strumenti sono ammessi, per quali attività e quali dati possono essere trasferiti a servizi esterni (capitolo 6);

- ruoli e responsabilità assegnati, con un titolare del rischio per ciascun sistema;
- processi periodici di valutazione e audit, collegati al golden set (capitolo 14);
- misure per sostenere lo sviluppo dell'alfabetizzazione in materia di AI del personale, richieste dall'art. 4 dell'AI Act come modificato dal Regolamento (UE) 2026/1744;
- la documentazione che lega questi elementi.

In questo modo AI Act, GDPR, NIS2 e criteri etici diventano un sistema di regole verificabile, in cui ogni strumento AI del SOC ha uno scopo definito e un responsabile. Il project work impiega insieme gli elementi della governance trattati nel corso, alcuni dei quali nelle letture di raccordo: i playbook (capitolo 1), l'audit trail dell'automazione (capitolo 5), la catena di custodia (capitolo 7), la spiegabilità (capitolo 8), il threat model dei sistemi AI (capitolo 13), l'osservabilità (capitolo 14).

Scenario didattico. Un'ispezione chiede conto dell'uso dell'AI nel SOC. L'organizzazione presenta inventario e classificazione dei sistemi, policy d'uso approvata, valutazioni d'impatto dei sistemi ad alto rischio, risultati periodici del golden set e registro delle decisioni automatizzate con gli interventi umani, documenti prodotti durante l'esercizio dei sistemi e non ricostruiti alla vigilia dell'ispezione.

15.10 Laboratorio: policy d'uso dell'AI per un SOC fittizio

La policy si redige in gruppi, con la guida del docente, a partire da un SOC fittizio assegnato a ciascun gruppo con la descrizione dei sistemi AI in uso, dei dati trattati e dell'organizzazione. Il lavoro comprende:

1. inventario dei sistemi AI e loro classificazione secondo l'AI Act, con motivazione;
2. matrice di responsabilità: chi approva l'adozione, chi revisiona gli output, chi risponde degli errori;
3. requisiti di registrazione degli eventi e di revisione umana per ciascun sistema;
4. discussione plenaria con revisione tra pari delle policy prodotte.

Consegna: policy di tre o quattro pagine, in forma adatta all'approvazione da parte di un dirigente; tabella di classificazione dei sistemi, motivata; una criticità lasciata aperta, con le opzioni possibili. La policy prodotta vale come vincolo nello scenario del project work (capitolo 16).

15.11 Letture

- Regolamento (UE) 2024/1689 (AI Act), in particolare artt. 3, 4, 5, 6, 12, 14, 25, 26, 50, 99 e allegato III, nel testo modificato dal Regolamento (UE) 2026/1744 (Omnibus digitale sull'IA).
- Commissione europea, *Guidelines on prohibited artificial intelligence practices established by Regulation (EU) 2024/1689 (AI Act)*, febbraio 2025.
- Regolamento (UE) 2016/679 (GDPR), in particolare artt. 5, 6, 22, 33–35 e capo V.

- Gruppo di lavoro Articolo 29, *Linee guida sul processo decisionale automatizzato relativo alle persone fisiche e sulla profilazione* (WP251 rev.01), 2018, fatte proprie dall'EDPB.
- Corte di giustizia dell'Unione europea, sentenza del 7 dicembre 2023, causa C-634/21 (SCHUFA Holding).
- Direttiva (UE) 2022/2555 (NIS2) e D.Lgs. 4 settembre 2024, n. 138, in particolare artt. 25 e 38.
- L. 20 maggio 1970, n. 300 (Statuto dei lavoratori), art. 4; L. 23 settembre 2025, n. 132, art. 11.
- Direttiva (UE) 2024/2853 sulla responsabilità per danno da prodotti difettosi.
- NIST AI 100-1, *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*, 2023.
- L. Floridi et al., *AI4People. An Ethical Framework for a Good AI Society: Opportunities, Risks, Principles, and Recommendations*, Minds and Machines, 28, 2018.

15.12 Domande di autovalutazione

1. In quali casi un sistema di sicurezza basato sull'AI ricade nell'alto rischio dell'AI Act?
Traccia. Quando rientra in un ambito dell'allegato III, per esempio se monitora e valuta il comportamento dei lavoratori o incide sulle decisioni relative al rapporto di lavoro. Se il sistema effettua profilazione di persone fisiche la deroga dell'art. 6, par. 3, non si applica. Secondo il considerando 55 i componenti usati esclusivamente per la cybersicurezza non dovrebbero essere considerati componenti di sicurezza delle infrastrutture critiche; la classificazione va fatta e documentata sistema per sistema.
2. Quando un contenimento automatico integra una decisione ai sensi dell'art. 22 GDPR?
Traccia. Quando la decisione è basata unicamente sul trattamento automatizzato e produce effetti giuridici o significativi analoghi sulla persona, per esempio la sospensione automatica dell'account di un dipendente con effetti sul rapporto di lavoro. Un intervento umano effettivo, con autorità e competenza per modificare la decisione, esclude l'applicazione della regola.
3. Chi risponde del danno causato da un blocco automatico errato? *Traccia.* Risponde l'organizzazione che impiega il sistema, come deployer, per l'uso, la configurazione e la scelta di automatizzare. Il fornitore risponde per il sistema come progettato e dichiarato, secondo il contratto e la disciplina sui prodotti, e l'operatore per le decisioni di sua competenza. Sulla documentazione preventiva si valuta la diligenza.
4. Quali dati di telemetria richiedono una DPIA? *Traccia.* Quelli trattati in un monitoraggio sistematico del comportamento delle persone, in particolare dei dipendenti (UEBA, DLP, rilevazione delle minacce interne), e i trattamenti effettuati con tecnologie innovative, compresi i sistemi di intelligenza artificiale, quando ricorre almeno un altro dei criteri del WP248 rev.01 (elenco del Garante, allegato 1, punto 7). L'elenco del Garante comprende i trattamenti che consentono il controllo a distanza dei lavoratori.
5. Perché la spiegabilità è anche un requisito probatorio? *Traccia.* Perché un alert o una decisione che ha avuto conseguenze su una persona deve poter essere motivato in

contraddittorio: senza la possibilità di indicare dati, criteri e versione del modello che hanno prodotto il risultato, l'esito non è difendibile in giudizio (capitolo 8).

Capitolo 16. Project work finale

Il capitolo descrive il project work che conclude il corso: la simulazione di un'intrusione in più fasi in un'azienda fittizia, gestita da gruppi che svolgono risposta e indagine forense con e senza il supporto dell'intelligenza artificiale. La prima parte illustra lo svolgimento della simulazione, le consegne richieste a ciascun gruppo, la rubrica di valutazione e la discussione finale sulle lezioni apprese. Gli approfondimenti trattano la costruzione dello scenario, la struttura del rapporto di incidente, la discussione orale, l'organizzazione dei ruoli nel gruppo, le condizioni per l'uso dell'AI e la revisione post-incidente. Il capitolo richiama, con rinvii puntuali, metodi e vincoli della traccia core e delle specializzazioni; ha quattro sezioni teoriche e nessun laboratorio separato, perché il laboratorio coincide con la simulazione.

Percorso. Traccia core, a conclusione del corso, in gruppi misti con un filone d'indagine per specializzazione (§ 16.8).

Obiettivi di apprendimento. Al termine del capitolo lo studente è in grado di:

- integrare metodologia di risposta agli incidenti, strumenti basati sull'AI e vincoli giuridici nella gestione di un caso simulato complesso;
- misurare e discutere i tempi di rilevamento e di risposta (MTTD e MTTR) ottenuti con e senza il supporto dell'AI;
- produrre una relazione forense completa e difendibile e condurre la revisione post-incidente.

16.1 Svolgimento della simulazione

La simulazione riprende lo scenario ransomware dell'esercitazione tabletop del capitolo 1 (§ 1.10), arricchito con log sintetici e con le evidenze necessarie a un'indagine tecnica completa, e si svolge in tre fasi:

- **tabletop:** il gruppo, formato da studenti delle tre specializzazioni, riceve lo scenario, assegna i filoni d'indagine e i ruoli di coordinamento (§ 16.8) e pianifica la risposta nel rispetto della policy d'uso dell'AI redatta nel laboratorio del capitolo 15 (§ 15.10), che nella simulazione vale come vincolo: un'azione che la viola è penalizzata anche se tecnicamente efficace;
- **laboratorio:** il gruppo conduce l'indagine su materiale predisposto (log del SIEM, dump di memoria, traffico di rete), mappa le tecniche osservate sulla matrice MITRE ATT&CK, propone e motiva le azioni di contenimento; metà dei gruppi dispone di strumenti AI (un copilota basato su un modello locale e il triage automatico degli alert basato su modelli), metà lavora senza, mentre Timesketch con i suoi analyzer, automatismi di arricchimento della timeline in gran parte basati su regole, è disponibile in entrambe le condizioni;
- **restituzione:** i gruppi presentano le relazioni, confrontano le metriche raccolte, conducono la discussione sulle lezioni apprese (§ 16.4) e si scambiano i documenti per una revisione tra pari.

Il confronto tra gruppi con e senza AI costituisce la parte sperimentale del corso. Perché il confronto sia interpretabile occorrono alcune condizioni:

- definizioni operative fissate prima della simulazione: per esempio, MTTD come intervallo tra l'inizio della fase di laboratorio e la dichiarazione motivata dell'incidente, MTTR come intervallo tra la dichiarazione e il completamento del contenimento; la sigla MTTR ha più significati (§ 1.3) e va accompagnata dalla definizione adottata;
- misure raccolte da un osservatore assegnato a ciascun gruppo, che registra l'orario delle tappe principali senza intervenire nel lavoro;
- indicatori ulteriori rispetto ai tempi: tecniche ATT&CK individuate correttamente, errori commessi, evidenze trascurate;
- assegnazione dei gruppi alle due condizioni per sorteggio, per limitare l'effetto della diversa composizione dei gruppi.

Ogni gruppo gestisce un solo incidente, quindi ogni misura è un singolo intervallo e la media si calcola sui gruppi della stessa condizione. Con pochi gruppi per condizione le differenze osservate hanno valore indicativo. La discussione serve a interpretarle alla luce di ciò che è accaduto nei singoli gruppi, per esempio un vantaggio di velocità annullato da un indicatore errato accettato senza verifica.

16.2 Consegne del gruppo

Le consegne principali sono tre documenti, destinati ai lettori tipici della documentazione di un incidente: il tecnico o il perito che ne verificherà il contenuto, il vertice aziendale che deve prendere decisioni, l'autorità che riceve le notifiche o che può chiedere conto delle scelte compiute, e la capacità di scrivere per ciascun destinatario rientra nella valutazione.

La **relazione forense** ricostruisce la cronologia dell'attacco mappandola sulle tecniche ATT&CK, presenta le evidenze con la relativa catena di custodia (capitolo 7), contiene una sezione metodologica che dichiara l'eventuale uso dell'AI (§ 16.9) e si chiude con le raccomandazioni. La struttura dettagliata è descritta nel § 16.6.

L'**executive summary**, spesso il primo documento letto da chi valuta l'operato del gruppo, è un documento di una pagina destinato al vertice aziendale: che cosa è accaduto, quale impatto ha avuto, che cosa è stato fatto, che cosa resta da fare. Il linguaggio è non tecnico e ogni affermazione deve corrispondere a un contenuto della relazione.

Il **registro delle decisioni** riporta ogni decisione con data e ora, motivazione, informazioni disponibili in quel momento e punto di approvazione umana. Comprende le valutazioni degli obblighi di notifica, i cui termini decorrono dal momento in cui l'organizzazione viene a conoscenza dei fatti: la violazione di dati personali ai sensi degli artt. 33 e 34 GDPR, la cui notifica spetta al titolare, e l'incidente significativo ai sensi della disciplina NIS2 (capitolo 15). Va motivata anche la decisione di non notificare: l'art. 33, par. 5, GDPR impone di documentare qualsiasi violazione di dati personali, anche se non notificata.

Ogni filone d'indagine (§ 16.8) aggiunge alla relazione un allegato:

- **rilevamento e risposta:** azioni di contenimento con il livello di autonomia di ciascuna e una regola di detection proposta, con il piano di test;
- **evidenze:** registro delle acquisizioni, valori di hash e comandi che rendono riproducibile l'analisi;
- **sistemi AI:** ricostruzione dei fatti dall'audit trail dell'assistente AI interno e casi proposti per il golden set.

A questi documenti si aggiunge una breve revisione post-incidente, descritta nel § 16.10. Il termine di consegna è fissato dal docente.

16.3 Rubrica di valutazione del project work

Il project work vale il 40% del voto finale del corso; la parte restante si compone della prova scritta (30%), dei laboratori (20%) e della partecipazione ai casi di studio (10%). La prova scritta ha una parte comune sui capitoli 1, 2, 7 e 15 e una parte sui capitoli della specializzazione, di pari peso. Il voto dei laboratori è la media di quelli dei moduli seguiti. Il lavoro è valutato con una rubrica analitica: cinque criteri, ciascuno con un peso e con quattro livelli (insufficiente, sufficiente, buono, eccellente), i cui descrittori espliciti sono consegnati agli studenti prima della simulazione.

I criteri e i pesi sono i seguenti:

- correttezza tecnica dell'indagine (30%), valutata per filone: catena dell'attacco ricostruita, evidenze interpretate correttamente;
- rigore metodologico e catena di custodia (25%): rispetto del processo forense descritto nel capitolo 7, dall'acquisizione alla documentazione;
- uso critico e documentato dell'AI (20%): impiego dichiarato, output verificati, errori del modello riconosciuti;
- qualità della relazione e dell'executive summary (15%): chiarezza, completezza, adeguatezza al destinatario;
- lavoro di gruppo e gestione dei ruoli (10%): ruoli esercitati, decisioni tracciate nel registro.

Per la correttezza tecnica ogni studente riceve il punteggio del filone che ha seguito, mentre gli altri quattro criteri si valutano sul lavoro del gruppo. Per l'incident commander, che coordina l'analisi senza svolgerla, vale la media dei filoni del gruppo.

La correttezza tecnica ha il peso maggiore e il rigore metodologico le si avvicina, perché una diagnosi corretta sostenuta da evidenze acquisite senza catena di custodia perde gran parte del suo valore quando il rapporto viene verificato da terzi. L'uso dell'AI è valutato per la qualità del controllo esercitato sugli output: su questo criterio un gruppo che usa estesamente l'AI senza verificarne i risultati ottiene un punteggio inferiore a quello di un gruppo che la usa poco e documenta ogni passaggio.

Esempio. Per il criterio sull'uso dell'AI i descrittori possono distinguere quattro situazioni, una per livello. È insufficiente un lavoro con output inseriti senza verifica o con uso non dichiarato, sufficiente un uso dichiarato con verifiche parziali. Il livello buono richiede verifiche documentate per ogni output rilevante, quello eccellente anche la discussione degli errori del modello come parte dell'analisi.

16.4 Lezioni apprese e discussione finale

La restituzione si conclude con una discussione che chiude il ciclo PICERL introdotto nel capitolo 1: le lezioni apprese sono una fase del processo di risposta e producono modifiche alla preparazione. La discussione si svolge in plenaria e riguarda i risultati e il processo.

Sui risultati si esamina dove i gruppi con AI sono stati più rapidi e dove sono stati fuorviati, quali evidenze hanno determinato la ricostruzione della catena dell'attacco, se le azioni di contenimento erano proporzionate e reversibili. Il confronto tra le due condizioni rende osservabili gli effetti descritti nel corso: l'accelerazione delle attività di sintesi e di ricerca, e insieme i rischi dell'automation bias (capitolo 5) e degli indicatori inventati dal modello (capitolo 6).

Sul processo si valuta se la relazione reggerebbe un contraddittorio tecnico, se la policy d'uso dell'AI ha funzionato come vincolo utile oppure come ostacolo e per quali ragioni, quali lacune sono emerse e che cosa andrebbe modificato nella preparazione. Le risposte all'ultimo punto si traducono in interventi su playbook, strumenti e formazione.

La discussione segue i criteri della revisione blameless descritta nel § 16.10: l'analisi riguarda le decisioni e le condizioni in cui sono state prese, senza attribuire colpe individuali. La revisione tra pari completa la fase: ogni gruppo esamina la relazione di un altro gruppo con i criteri della rubrica e ne segnala i punti deboli, in particolare le affermazioni prive di evidenza.

16.5 Struttura dello scenario della simulazione

La gestione di un incidente richiede insieme, sotto pressione e con informazioni incomplete, le competenze esercitate separatamente nella traccia core e nelle specializzazioni. Lo scenario del project work è un'intrusione con ransomware ed esfiltrazione di dati in un'azienda fittizia, costruito come esercitazione a **inject**: il quadro emerge progressivamente dalle evidenze (alert, dump di memoria, log di rete, timeline) e da eventi introdotti durante lo svolgimento, tra cui una richiesta della funzione legale e una richiesta di dichiarazioni da parte della stampa.

Ogni inject mette alla prova la competenza di uno o due capitoli. Il dump di memoria richiede le tecniche di analisi del capitolo 9. La richiesta della funzione legale di valutare la notifica ai sensi dell'art. 33 GDPR richiama il capitolo 15. Un messaggio attribuito al vertice aziendale, che chiede di riattivare subito i sistemi, va trattato con le cautele sui contenuti sintetici e sull'ingegneria sociale del capitolo 12, verificando l'identità del mittente su un canale indipendente prima di qualsiasi azione. Un'istruzione nascosta in un ticket ed eseguita dall'assistente AI in-

terno dell'azienda richiede l'analisi della prompt injection indiretta (§ 13.6) e la ricostruzione dei fatti dall'audit trail dell'assistente (§ 14.6). Nei gruppi senza strumenti AI gli ultimi due inject restano, perché riguardano l'AI dell'attaccante e l'assistente dell'azienda.

Il gruppo opera con i ruoli definiti nel capitolo 1; nella condizione con AI decide anche quando usarla e quando evitarla, e la scelta è oggetto di valutazione insieme alle sue conseguenze.

Errori frequenti. Il gruppo si concentra sull'analisi tecnica e trascura il processo: registro delle decisioni assente, catena di custodia improvvisata, comunicazione rinviata. Nella rubrica il rigore metodologico ha un peso vicino a quello della correttezza tecnica (§ 16.3), perché chi esamina un rapporto dall'esterno ne verifica anche il processo.

16.6 Struttura del rapporto di incidente

Il rapporto resta agli atti dopo l'incidente e lo leggono il vertice aziendale, la funzione legale, l'assicuratore e, in alcuni casi, l'autorità giudiziaria. La struttura attesa comprende:

- un **executive summary** di una pagina in linguaggio non tecnico, consegnato anche come documento autonomo (§ 16.2): fatti, impatto, stato attuale, decisioni richieste;
- una timeline ricostruita, con la fonte di ciascun evento (log, artefatto, dichiarazione) e il fuso orario di riferimento;
- un'analisi tecnica con evidenze citate e riproducibili: valori hash, riferimenti agli allegati, strumenti e versioni impiegati;
- le azioni intraprese, collegate alle voci del registro delle decisioni;
- raccomandazioni ordinate per priorità e costo;
- un'appendice metodologica sull'uso dell'AI (§ 16.9).

Ogni affermazione deve poter essere ricondotta a un'evidenza, secondo lo standard della relazione tecnica forense (capitolo 7). Le conclusioni vanno graduate in base alla solidità delle evidenze: un fatto accertato, un'ipotesi compatibile con le evidenze e un'ipotesi da verificare si esprimono con formule diverse. La relazione dichiara inoltre i limiti dell'analisi, per esempio le fonti di log mancanti o i sistemi che non è stato possibile acquisire.

L'uso dell'AI va dichiarato nell'appendice metodologica, la cui completezza è valutata positivamente. Un IoC prodotto dal modello e non verificato (capitolo 6) è un errore grave, perché introduce nel documento un'affermazione priva di base che un lettore tecnico può smentire con una ricerca nei log.

Metodo. L'executive summary si verifica facendolo leggere a una persona estranea al gruppo, che al termine della lettura deve saper dire che cosa è accaduto, quanto è grave e che cosa va deciso. Se servono spiegazioni sui termini, il testo va riscritto.

16.7 Discussione orale del project work

La discussione finale è impostata come un **controesame tecnico**, sul modello delle domande rivolte in udienza a un consulente tecnico (capitolo 8), e verifica la capacità di sostenere ciò che è scritto nel rapporto. Le domande tipiche riguardano la scelta di contenere un host e non un al-

tro, l'evidenza su cui poggia una conclusione e la sua tenuta se contestata, il modo in cui è stato verificato l'output di un LLM, la decisione che il gruppo prenderebbe oggi con le informazioni acquisite in seguito. Ogni studente risponde a una domanda sul filone della propria specializzazione e a una su un filone di un'altra; la seconda verifica che i risultati siano stati condivisi nel gruppo.

Una risposta solida espone la catena tra evidenza, ragionamento e decisione, insieme ai limiti dell'analisi e a ciò che al momento della decisione era sconosciuto. L'ultima domanda dell'elenco riguarda la distorsione del senno di poi (hindsight bias): una decisione va giudicata in base alle informazioni disponibili quando è stata presa, e quelle informazioni si ricostruiscono dal registro delle decisioni, che ne documenta data, ora e motivazione.

Metodo. Prima della discussione il rapporto va riletto dal punto di vista di un consulente tecnico della controparte: ogni affermazione priva di un'evidenza o di un riferimento va corretta o eliminata. Per ogni conclusione principale conviene preparare l'elenco delle evidenze che la sostengono e delle ipotesi alternative considerate e scartate, con il motivo dell'esclusione.

Errori frequenti. Difendere a oltranza una conclusione debole senza riconoscerne e analizzarne i limiti. Un altro errore è attribuire all'AI una conclusione del gruppo, la cui responsabilità resta di chi firma il rapporto.

16.8 Ruoli e coordinamento del gruppo

Il project work valuta anche il funzionamento del gruppo. I ruoli introdotti nel capitolo 1, ai quali si aggiunge quello di scribe, vanno assegnati all'inizio e mantenuti per tutta la simulazione:

- l'**incident commander** (responsabile della gestione) decide, assegna i compiti e protegge il gruppo dalle interruzioni, coordinando l'analisi senza svolgerla;
- gli analisti seguono filoni d'indagine paralleli, uno per specializzazione: rilevamento e risposta, evidenze, sistemi AI;
- il referente per la comunicazione prepara gli aggiornamenti per il vertice e le risposte alle richieste esterne;
- il referente legale valuta gli obblighi di notifica e i vincoli sul trattamento dei dati;
- lo **scriba** registra decisioni e orari e tiene il registro delle decisioni su cui si basano il rapporto (§ 16.6) e la discussione (§ 16.7).

L'organizzazione per ruoli deriva dall'Incident Command System, nato per la gestione delle emergenze; il volume *Site Reliability Engineering* ne descrive l'applicazione agli incidenti dei servizi informatici. Il coordinamento si basa su brevi punti di sincronizzazione a cadenza fissa, in cui ogni filone riassume che cosa ha trovato, che cosa sta facendo e di che cosa ha bisogno.

Scenario didattico. Nel gruppo A tutti lavorano sul dump di memoria, nessuno registra le decisioni e l'incident commander analizza invece di coordinare. I risultati si duplicano, la timeline resta incompleta e, quando arriva la richiesta della stampa, nessun aggiornamento è pronto. Il gruppo B, con le stesse competenze tecniche, produce un rapporto coerente grazie a ruoli chiari, sincronizzazioni regolari e uno scriba attivo.

Errori frequenti. L'incident commander che si dedica all'analisi lascia scoperto il coordinamento quando la visione d'insieme è più necessaria, per esempio quando arrivano insieme un nuovo indicatore e una richiesta esterna.

16.9 Uso dichiarato e verificato dell'AI

Dove l'AI è ammessa, il suo impiego è subordinato alla **dichiarazione** e alla **verifica**. Un'appendice metodologica dichiara quali strumenti sono stati usati, in quali passaggi, con quali modelli e versioni e con quali controlli umani, e ogni output dell'AI che entra nel rapporto deve essere stato riscontrato sul dato originale. Le forme di verifica dipendono dal tipo di output:

- un IoC indicato da un riassunto si cerca nei log e negli artefatti da cui il riassunto è stato prodotto;
- una timeline ricostruita dal modello si confronta, evento per evento, con le fonti;
- la funzione di uno script spiegata da un LLM si conferma con l'analisi dinamica in sandbox (capitolo 11);
- una mappatura su ATT&CK proposta dal modello si controlla sulla descrizione della tecnica pubblicata da MITRE.

La valutazione considera l'uso consapevole: impiegare l'AI dove accelera il lavoro (sintesi, bozze, scrittura di query) e tenerla fuori dai verdetti e dalle decisioni di contenimento, che restano al gruppo. L'appendice serve anche alla riproducibilità, perché chi riesamina il caso deve poter sapere quale parte del rapporto è stata prodotta o assistita da un modello e con quali controlli.

Scenario didattico. Un gruppo usa l'LLM per la bozza dell'executive summary e per spiegare uno script offuscato, verifica entrambi i risultati e lo dichiara nell'appendice. Un altro gruppo inserisce l'output del modello come analisi definitiva, compresi IoC assenti dai log, e sul criterio dell'uso dell'AI ottiene una valutazione inferiore.

Cautele. I dati dello scenario sono trattati come evidenze d'indagine e valgono le regole del capitolo 6: nessun dato riservato verso servizi esterni non autorizzati, modello locale per i contenuti sensibili. Il rispetto del perimetro dei dati rientra nella valutazione, insieme alla conformità alla policy del capitolo 15.

16.10 Revisione post-incidente e miglioramento

Un incidente si conclude con le **lezioni apprese**. La SP 800-61 Rev. 2 del NIST indicava questa attività come una delle parti più importanti della risposta e insieme come quella omessa più spesso. La Rev. 3 riconduce il miglioramento alla categoria Improvement (ID.IM) della funzione Identify del Cybersecurity Framework 2.0, che raccoglie i miglioramenti individuati in tutte le funzioni.

La revisione post-incidente condotta senza ricerca del colpevole (**blameless**) individua le cause sistemiche e produce azioni con responsabile e scadenza. L'analisi riguarda il modo in cui è avvenuto l'accesso iniziale e come si chiude, il controllo che è mancato, la detection da aggiungere secondo il ciclo descritto nel capitolo 3, le pratiche che hanno funzionato e vanno consolidate. L'impostazione blameless, descritta tra l'altro nel capitolo sulla cultura dei postmortem del volume *Site Reliability Engineering*, favorisce una ricostruzione completa: chi teme una sanzione tende a tacere i propri errori, che sono spesso le informazioni più utili.

La revisione alimenta anche il miglioramento dei sistemi AI difensivi. Ogni caso gestito entra nel golden set con cui si valutano i sistemi (capitolo 14), e gli errori di un copilota, per esempio un indicatore inventato o una tecnica attribuita erroneamente, diventano test di regressione da eseguire prima di ogni aggiornamento del modello o della configurazione.

Metodo. La consegna del project work comprende una breve revisione post-incidente: tre cause sistemiche, tre azioni con responsabile e scadenza, una nuova regola di detection derivata dall'episodio e provata sui log dello scenario.

Ruolo dell'AI. Gli strumenti AI accelerano la stesura della timeline e la ricerca di casi simili nello storico; le cause sistemiche si individuano nel confronto tra i ruoli coinvolti.

16.11 Letture

- NIST, *SP 800-61 Rev. 3, Incident Response Recommendations and Considerations for Cybersecurity Risk Management: A CSF 2.0 Community Profile*, aprile 2025.
- NIST, *SP 800-61 Rev. 2, Computer Security Incident Handling Guide*, 2012 (ritirata il 3 aprile 2025 e sostituita dalla Rev. 3), sezione 3.4 sull'attività post-incidente.
- NIST, *SP 800-84, Guide to Test, Training, and Exercise Programs for IT Plans and Capabilities*, 2006.
- P. Kral, *Incident Handler's Handbook*, SANS Institute, 2012.
- B. Beyer, C. Jones, J. Petoff, N. R. Murphy (a cura di), *Site Reliability Engineering: How Google Runs Production Systems*, O'Reilly, 2016, capitoli 14 (*Managing Incidents*) e 15 (*Postmortem Culture: Learning from Failure*).
- MITRE, *ATT&CK for Enterprise*, matrice e schede delle tecniche.
- ISO/IEC 27037:2012, *Guidelines for identification, collection, acquisition and preservation of digital evidence*.

16.12 Domande di autovalutazione

1. Perché il confronto tra gruppi con e senza AI richiede definizioni di MTTD e MTTR fissate prima della simulazione? *Traccia.* La sigla MTTR ha più significati e gli istanti di inizio e di fine di ogni intervallo vanno stabiliti in anticipo, altrimenti ogni gruppo misura una grandezza diversa. Le definizioni, applicate da un osservatore, rendono confrontabili i gruppi; con pochi gruppi le differenze restano indicative e vanno lette insieme agli errori e alle tecniche individuate.
2. Quali informazioni deve contenere il registro delle decisioni e perché vi si annota anche la decisione di non notificare? *Traccia.* Data e ora, decisione, motivazione, informazioni disponibili in quel momento, persona che ha approvato. La decisione di non notificare è una valutazione giuridica che può essere contestata: l'art. 33, par. 5, GDPR impone di documentare qualsiasi violazione di dati personali, e il registro dimostra che gli obblighi GDPR e NIS2 sono stati esaminati entro i termini.
3. Come si verifica un IoC proposto da un LLM prima di inserirlo nel rapporto? *Traccia.* Si cerca l'indicatore nei log e negli artefatti originali da cui il modello ha tratto il riassunto; se non compare, si scarta. La verifica e il suo esito si annotano nell'appendice metodologica. Un IoC non verificato è un errore grave perché introduce nel rapporto un'affermazione priva di base (capitolo 6).
4. Quali compiti ha l'incident commander e perché non conduce direttamente l'analisi? *Traccia.* Decide, assegna i compiti, protegge il gruppo dalle interruzioni, guida le sincronizzazioni e cura il raccordo con comunicazione e funzione legale. Se si dedica all'analisi perde la visione d'insieme e lascia scoperte le decisioni di coordinamento nel momento in cui arrivano nuovi indicatori o richieste esterne.
5. Che cosa caratterizza una revisione post-incidente blameless e quali prodotti deve generare? *Traccia.* Analizza le decisioni e le condizioni in cui sono state prese senza attribuire colpe individuali, così che errori e omissioni emergano. Produce cause sistemiche, azioni con responsabile e scadenza, nuove regole di detection, nuovi casi per il golden set e test di regressione per i sistemi AI.

Glossario

- **Accertamento tecnico non ripetibile:** accertamento su persone, cose o luoghi il cui stato è soggetto a modificazione, anche per effetto dell'accertamento stesso, e che perciò non può essere ripetuto nelle stesse condizioni; se disposto dal pubblico ministero è soggetto alle garanzie dell'art. 360 c.p.p.
- **Accuratezza robusta:** accuratezza di un modello misurata su input modificati da un avversario entro limiti stabiliti, in contrapposizione all'accuratezza su dati ordinari.
- **Acquisizione fisica e logica:** l'acquisizione fisica è la copia bit per bit dell'intero supporto, compreso lo spazio non allocato; l'acquisizione logica copia file, cartelle o partizioni selezionati.
- **Acquisizione logica:** estrazione dei dati che il sistema operativo di un dispositivo espone tramite interfacce e backup.
- **Admiralty Code:** sistema di valutazione che assegna una lettera (A–F) all'affidabilità della fonte e un numero (1–6) alla credibilità dell'informazione.
- **AFU e BFU** (After First Unlock, Before First Unlock): stati di un dispositivo mobile dopo il primo sblocco successivo all'avvio, con molte chiavi di cifratura disponibili in memoria, o prima di esso, con la maggior parte dei dati inaccessibile.
- **Agente (basato su LLM):** modello linguistico inserito in un ciclo di pianificazione ed esecuzione che invoca strumenti esterni con effetti sui sistemi.
- **AI Act:** Regolamento (UE) 2024/1689, che disciplina i sistemi di intelligenza artificiale secondo un approccio basato sul rischio.
- **Alert:** evento o combinazione di eventi che una regola o un modello segnala come potenzialmente rilevante per la sicurezza.
- **Alert fatigue:** calo dell'attenzione degli analisti causato da un volume elevato di alert, in gran parte falsi positivi.
- **Alfabetizzazione in materia di AI:** obbligo per fornitori e deployer di adottare misure per sostenere lo sviluppo dell'alfabetizzazione in materia di AI del proprio personale, senza obbligo di garantirne un livello determinato (art. 4 AI Act, come modificato dal Reg. (UE) 2026/1744).
- **Allucinazione:** affermazione plausibile prodotta da un modello linguistico ma priva di fondamento nei dati o nei fatti; il profilo NIST AI 600-1 la indica come confabulation.
- **Anomaly detection:** tecnica che segnala gli elementi che si discostano da una norma appresa dai dati, senza richiedere esempi etichettati di attacco.
- **Appendice metodologica:** sezione del rapporto di incidente che dichiara gli strumenti AI impiegati, i passaggi in cui sono stati usati, modelli e versioni e i controlli umani effettuati.
- **Apprendimento non supervisionato:** paradigma in cui il modello individua strutture nei dati senza etichette, per esempio gruppi o anomalie.

- **Apprendimento per rinforzo:** paradigma in cui un agente apprende una politica di azione sulla base delle ricompense ricevute da un ambiente.
- **Apprendimento supervisionato:** paradigma in cui il modello apprende da esempi etichettati una funzione che associa le caratteristiche all'etichetta.
- **Atomic Red Team:** libreria open source di test che riproducono singole tecniche ATT&CK per verificare le detection.
- **Audit trail:** registro a sola aggiunta delle azioni eseguite da sistemi e operatori, con orario, base decisionale, autore ed esito.
- **Autoencoder:** rete neurale che impara a ricostruire i dati normali; un errore di ricostruzione elevato indica un'anomalia.
- **Automation bias:** tendenza ad affidarsi all'indicazione di un sistema automatico anche quando altre informazioni la contraddicono.
- **Backdoor:** comportamento nascosto inserito in un modello durante l'addestramento, che si attiva solo in presenza di un segnale scelto dall'attaccante (trigger).
- **Barriera d'ingresso:** livello di competenze e risorse necessario per condurre un attacco; l'AI generativa tende ad abbassarla.
- **Base rate fallacy:** errore che consiste nel trascurare la frequenza di base di un fenomeno nell'interpretare un risultato; nella detection porta a sovrastimare l'affidabilità degli alert su eventi rari, il cui valore predittivo è basso per effetto della bassa prevalenza.
- **Baseline comportamentale:** modello del comportamento normale di un'entità, rispetto al quale si misurano gli scostamenti.
- **Beaconing:** contatto periodico di un impianto malevolo con il server di comando e controllo, riconoscibile dalla regolarità degli intervalli e delle dimensioni.
- **BEC (Business Email Compromise):** frode in cui l'attaccante compromette o imita una casella di posta aziendale per deviare pagamenti.
- **Blameless** (revisione senza ricerca del colpevole): impostazione della revisione post-incidente che analizza decisioni e cause sistemiche senza attribuire colpe individuali.
- **C2PA (Coalition for Content Provenance and Authenticity):** standard che associa a un file multimediale un manifesto firmato con l'origine e le modifiche dichiarate.
- **CACAO (Collaborative Automated Course of Action Operations):** specifica OASIS (Committee Specification) per la descrizione dei playbook di sicurezza in formato JSON indipendente dal prodotto.
- **Carving:** tecnica di recupero dei file a partire dal contenuto, riconoscendo intestazioni e strutture dei formati.
- **Catena di custodia:** documentazione ininterrotta dei passaggi di un'evidenza, con autori, date, luoghi, motivi e hash.
- **Chunking:** suddivisione dei documenti in frammenti da indicizzare per il retrieval.
- **Circuit breaker:** meccanismo che interrompe un flusso automatico quando il numero o la frequenza delle azioni supera una soglia prefissata.

- **Clustering di malware:** raggruppamento dei campioni in famiglie sulla base di misure di similarità del file, del codice, del comportamento e dell'infrastruttura.
- **Concept drift:** cambiamento nel tempo della distribuzione dei dati rispetto a quella di addestramento, con conseguente calo delle prestazioni del modello.
- **Consenso OAuth:** autorizzazione concessa a un'applicazione di terze parti ad accedere ai dati di un account, che può restare valida anche dopo il cambio della password.
- **Controesame tecnico:** discussione in cui un interlocutore verifica, con domande puntuali, la tenuta delle conclusioni di un rapporto e delle evidenze su cui si basano.
- **Copilota investigativo:** sistema che combina un modello linguistico con connettori verso i dati di sicurezza dell'organizzazione per assistere l'analista.
- **Criteri Daubert:** criteri indicati dalla Corte suprema degli Stati Uniti nel 1993 per vagliare l'affidabilità del metodo su cui si fonda la testimonianza dell'esperto: verificabilità, peer review, tasso di errore, accettazione nella comunità scientifica.
- **CSIRT (Computer Security Incident Response Team):** gruppo che coordina la gestione degli incidenti dichiarati.
- **Curva ROC:** curva del tasso di veri positivi in funzione del tasso di falsi positivi al variare della soglia; l'area sottesa (AUC) misura la capacità di separare le classi.
- **Database vettoriale:** archivio di embedding interrogabile per similarità.
- **Decisione basata unicamente sul trattamento automatizzato:** decisione presa senza un intervento umano effettivo, disciplinata dall'art. 22 GDPR.
- **Decompiler:** strumento che ricostruisce, a partire dal codice macchina, uno pseudocodice in un linguaggio di alto livello simile al C.
- **Deepfake:** contenuto audio o video sintetico che imita l'aspetto o la voce di una persona reale.
- **Defanging:** modifica di indicatori come URL e domini (per esempio hxxp, [..]) per impedire che diventino collegamenti attivi.
- **Deployer:** soggetto che utilizza un sistema AI sotto la propria autorità, salvo l'uso nel corso di un'attività personale non professionale (art. 3, punto 4, AI Act).
- **Detection engineering:** disciplina che progetta, verifica, rilascia e mantiene le detection come software.
- **Detection-as-code:** gestione delle detection come codice versionato, con revisione e test automatici.
- **DGA (Domain Generation Algorithm):** algoritmo con cui il malware genera domini pseudocasuali da contattare, rendendo inefficace il blocco tramite liste.
- **DMARC:** protocollo con cui il titolare di un dominio pubblica una politica che indica ai destinatari come trattare i messaggi per i quali né SPF né DKIM sono validi e allineati al dominio del campo From, e con cui riceve rapporti sugli esiti; non protegge da domini simili ma distinti.
- **DNS tunneling:** tecnica che trasporta dati all'interno di query e risposte DNS.

- **Doppia estorsione:** modello di attacco ransomware in cui i dati vengono esfiltrati prima della cifratura e la minaccia di pubblicazione si aggiunge alla richiesta di riscatto.
- **DPIA** (valutazione d'impatto sulla protezione dei dati): analisi preventiva dei rischi di un trattamento per i diritti delle persone, prevista dall'art. 35 GDPR.
- **Drift avversario:** concept drift causato dagli attaccanti, che modificano le proprie tecniche per eludere il modello.
- **Dwell time:** intervallo tra l'accesso iniziale dell'attaccante e la scoperta dell'intrusione.
- **Eccesso di autonomia** (excessive agency): rischio di un'applicazione LLM che dispone di strumenti, permessi o margini di decisione oltre il necessario (OWASP LLM06).
- **Embedding:** rappresentazione di un testo come vettore numerico, costruita in modo che testi di significato simile corrispondano a vettori vicini.
- **Entropia (di Shannon):** misura della casualità dei dati; nelle sezioni di un eseguibile valori vicini a 8 bit per byte indicano contenuti compressi o cifrati.
- **Error Level Analysis** (ELA): tecnica che ricomprime un'immagine JPEG e confronta i livelli di errore per evidenziare regioni con una storia di compressione diversa.
- **Esempio avversario** (adversarial example): input con modifiche mirate che cambiano la decisione di un modello senza alterarne la sostanza o, per il malware, il comportamento.
- **Evasion:** attacco in fase di inferenza che usa esempi avversari per eludere un classificatore già addestrato.
- **Executive summary:** sintesi di una pagina, in linguaggio non tecnico, destinata al vertice aziendale: fatti, impatto, stato della risposta, decisioni richieste.
- **EXIF:** formato dei metadati delle immagini, che può includere data, modello della fotocamera e coordinate GPS.
- **Explainable AI:** insieme di tecniche, come SHAP, LIME e le mappe di salienza, che indicano quali caratteristiche dell'input hanno inciso su una previsione.
- **F1:** media armonica di precision e recall.
- **False flag:** indizio inserito deliberatamente per indurre un'attribuzione errata.
- **Feature engineering:** trasformazione dei dati grezzi in variabili numeriche utilizzabili da un modello.
- **FIDO2:** insieme di standard per l'autenticazione basata su chiavi crittografiche legate al dominio, resistente al phishing.
- **Fine-tuning:** ulteriore addestramento di un modello pre-addestrato su dati di un dominio, che ne modifica i pesi.
- **Forensic readiness:** predisposizione di un sistema alla raccolta delle evidenze prima che si verifichi un incidente.
- **Formato PE** (Portable Executable): formato dei file eseguibili e delle librerie dei sistemi Windows.
- **Fornitore:** soggetto che sviluppa un sistema AI, o lo fa sviluppare, e lo immette sul mercato o lo mette in servizio con il proprio nome o marchio, anche per uso proprio (art. 3, punti 3 e 11, AI Act).

- **Freelist**: elenco delle pagine libere di un database SQLite, che possono contenere record cancellati non ancora sovrascritti.
- **Funzione hash crittografica**: funzione che produce un'impronta di lunghezza fissa per la quale è computazionalmente impraticabile trovare un input con un'impronta data o due input con la stessa impronta; la modifica di un bit dell'input ne cambia il valore in modo imprevedibile.
- **Fuzzy hashing**: famiglia di funzioni hash che misurano la somiglianza tra file invece della loro identità; ne sono esempi ssdeep e TLSH.
- **Golden set**: insieme di domande e casi con risposta corretta nota, usato per valutare un sistema AI e rilevarne le regressioni.
- **Groundedness**: misura in cui le affermazioni di un output sono riconducibili alle evidenze fornite.
- **Grounding**: ancoraggio della risposta di un modello alle evidenze fornite nel contesto.
- **Hash percettivo**: hash calcolato sull'aspetto visivo di un'immagine, stabile rispetto a ridimensionamento e ricomprensione.
- **Hindsight bias** (distorsione del senno di poi): tendenza a giudicare una decisione alla luce di informazioni acquisite dopo il momento in cui è stata presa.
- **Human-in-the-loop**: livello di autonomia in cui ogni azione attende un'approvazione umana esplicita.
- **Human-on-the-loop**: livello di autonomia in cui il sistema agisce sotto la supervisione di un operatore che può interrompere o annullare.
- **Human-out-of-the-loop**: livello di autonomia in cui il sistema agisce senza intervento umano.
- **Incident commander**: responsabile della gestione dell'incidente, che decide, assegna i compiti e coordina il gruppo senza svolgere direttamente l'analisi.
- **Incidente**: occorrenza che, senza autorizzazione legittima, mette in pericolo in modo effettivo o imminente riservatezza, integrità o disponibilità di informazioni o sistemi informativi, oppure costituisce una violazione, o una minaccia imminente di violazione, di leggi, policy, procedure di sicurezza o regole di uso accettabile (definizione NIST).
- **Ingegneria sociale**: insieme di tecniche che sfruttano la fiducia e il comportamento delle persone per ottenere accessi, informazioni o pagamenti.
- **Initial access broker**: soggetto che ottiene e rivende accessi iniziali alle reti delle organizzazioni.
- **Inject**: evento introdotto durante un'esercitazione per far evolvere lo scenario e mettere alla prova una specifica competenza.
- **IoC** (Indicator of Compromise): elemento osservabile, come un hash, un indirizzo IP o un dominio, associato a un'attività malevola.
- **ISAC** (Information Sharing and Analysis Center): comunità settoriale per lo scambio di informazioni sulle minacce.

- **Jitter**: variazione casuale introdotta negli intervalli tra i contatti di un beacon per renderli meno regolari.
- **Legittimo interesse**: base giuridica del trattamento prevista dall'art. 6, par. 1, lett. f, GDPR, applicabile alla sicurezza delle reti e dell'informazione nella misura strettamente necessaria e proporzionata (considerando 49), previo bilanciamento con i diritti degli interessati.
- **Liar's dividend**: possibilità di contestare come falso un contenuto autentico sfruttando la diffusione dei deepfake.
- **Live forensics**: acquisizione di memoria e stato di un sistema acceso, con una perturbazione documentata.
- **LLM** (Large Language Model): modello linguistico di grandi dimensioni addestrato a generare testo.
- **LLM-as-judge**: uso di un modello linguistico per valutare automaticamente gli output di un altro sistema.
- **Malware fileless**: malware che opera in memoria senza scrivere file eseguibili su disco.
- **Membership inference**: attacco alla riservatezza che stima se un determinato record faceva parte dei dati di addestramento di un modello.
- **MFA fatigue**: invio ripetuto di richieste di approvazione dell'autenticazione a più fattori nella speranza che l'utente ne approvi una.
- **MFT** (Master File Table): struttura di NTFS che contiene un record con metadati e timestamp per ogni file.
- **MISP**: piattaforma open source per la condivisione di threat intelligence basata su eventi, attributi e tag.
- **MITRE ATLAS**: base di conoscenza di tattiche, tecniche, mitigazioni e casi documentati di attacchi ai sistemi di intelligenza artificiale, strutturata sul modello di ATT&CK.
- **MITRE ATT&CK**: base di conoscenza delle tattiche e delle tecniche osservate in attacchi documentati.
- **MITRE D3FEND**: catalogo delle tecniche difensive, organizzato in relazione alle tecniche offensive.
- **Model extraction**: ricostruzione di una copia funzionale di un modello a partire dalle risposte a interrogazioni ripetute.
- **Modello open weight**: modello di cui sono pubblicati i pesi, eseguibile su infrastruttura propria.
- **MTTD** (Mean Time To Detect): tempo medio tra l'inizio degli incidenti e il loro rilevamento, calcolato su un insieme di casi.
- **MTTR**: tempo medio di risposta, ripristino o rimedio, secondo la definizione adottata dall'organizzazione.
- **NSRL** (National Software Reference Library): raccolta del NIST di hash di file appartenenti a software noto, usata per riconoscere e filtrare i file noti; la corrispondenza non ne attesta l'innocuità né l'irrilevanza per il caso.

- **Orchestrazione:** integrazione, tramite API, di strumenti di sicurezza diversi in un unico flusso di lavoro.
- **Ordine di volatilità:** sequenza di acquisizione dalla fonte più volatile alla più stabile, descritta nella RFC 3227.
- **Overfitting:** adattamento eccessivo del modello ai dati di addestramento, che ne riduce la capacità di generalizzare.
- **Packer:** programma che comprime o cifra un eseguibile e lo racchiude in uno stub che lo ricostruisce in memoria all'avvio.
- **Piramide del dolore:** modello che ordina gli indicatori secondo il costo che il loro rilevamento impone all'attaccante.
- **Playbook:** documento che definisce strategia, decisioni e responsabilità per una categoria di incidenti.
- **Poisoning:** contaminazione dei dati o del processo di addestramento per degradare un modello o alterarne le decisioni.
- **Policy d'uso dell'AI:** documento interno che stabilisce sistemi ammessi, dati trasferibili, revisioni umane obbligatorie, registrazione degli eventi e responsabilità.
- **Pre-notifica:** prima comunicazione al CSIRT Italia di un incidente significativo, da inviare entro 24 ore dalla conoscenza secondo il D.Lgs. 138/2024.
- **Pre-training:** addestramento iniziale di un modello linguistico su un corpus ampio e generale.
- **Precision:** quota di veri positivi tra gli elementi segnalati dal sistema.
- **Prefetch:** file di Windows che, quando il prefetching è attivo, registrano l'avvio degli eseguibili con numero e date recenti di esecuzione.
- **Privacy differenziale:** tecnica che limita in misura quantificabile l'influenza di ogni singolo record sul modello addestrato, riducendo il rischio di memorizzazione e di membership inference.
- **Privilegio minimo:** principio per cui ogni componente dispone solo dei permessi necessari al proprio compito.
- **PRNU (Photo Response Non-Uniformity):** impronta di rumore propria di ciascun sensore fotografico, usata per associare un'immagine a una fotocamera.
- **Process hollowing:** tecnica di iniezione che sostituisce in memoria l'immagine di un processo legittimo con codice malevolo.
- **Prompt injection indiretta:** inserimento di istruzioni ostili nei contenuti che un modello linguistico elabora, come e-mail, pagine web o documenti.
- **RaaS (Ransomware as a Service):** modello in cui gli sviluppatori di un ransomware lo offrono ad affiliati in cambio di un abbonamento o di una quota dei riscatti.
- **RAG (retrieval-augmented generation):** tecnica che a ogni richiesta recupera documenti pertinenti e li inserisce nel contesto del modello senza modificarne i pesi.
- **Recall:** quota di veri positivi tra gli elementi effettivamente positivi.

- **Red teaming:** verifica della resistenza di un sistema attraverso attacchi simulati, condotti con strumenti automatici e prove manuali.
- **Registro delle decisioni:** documento che riporta ogni decisione presa durante l'incidente con data e ora, motivazione, informazioni disponibili e approvazione umana.
- **Reranking:** riordino dei risultati del retrieval mediante un secondo modello che ne valuta la pertinenza rispetto alla domanda.
- **Ricerca ibrida:** combinazione di ricerca per similarità tra embedding e ricerca per parole chiave.
- **Ricerca semantica:** ricerca che restituisce i testi di significato vicino alla domanda, anche in assenza di parole in comune.
- **Richiamo (recall):** quota degli elementi rilevanti che un sistema di ricerca o di classificazione individua effettivamente.
- **Risk-based alerting:** approccio che accumula punteggi di rischio per entità e genera un alert solo al superamento di una soglia.
- **Rollback:** procedura che annulla gli effetti di un'azione di risposta.
- **Rubrica analitica:** strumento di valutazione che articola il giudizio in criteri pesati, ciascuno con livelli descritti esplicitamente.
- **Runbook:** procedura operativa passo per passo, eseguibile da un analista o da una piattaforma di automazione.
- **Safetensors:** formato per i pesi dei modelli che contiene esclusivamente tensori, per cui il caricamento non esegue codice.
- **Sandbox:** ambiente virtuale strumentato in cui un campione viene eseguito per osservarne il comportamento.
- **Scriba:** ruolo del gruppo di risposta che registra decisioni e orari durante la gestione dell'incidente.
- **Shellbag:** chiavi del registro di Windows che conservano informazioni sulle cartelle visualizzate dall'utente.
- **Sigma:** formato aperto per scrivere regole di detection indipendenti dal SIEM.
- **Sistema ad alto rischio:** sistema AI che rientra nell'art. 6 dell'AI Act e negli ambiti degli allegati I o III, soggetto ai requisiti degli artt. 9–15, che si applicano dal 2 dicembre 2027 per l'allegato III e dal 2 agosto 2028 per l'allegato I (Reg. (UE) 2026/1744).
- **Smear:** incoerenza di un dump di memoria dovuta alle modifiche delle pagine durante la copia.
- **SOAR (Security Orchestration, Automation and Response):** piattaforma che orchestra gli strumenti di sicurezza, automatizza i passi dei runbook e gestisce i casi.
- **SOC (Security Operations Center):** struttura che svolge monitoraggio continuo, gestione degli alert e primo triage.
- **Soggetti essenziali e importanti:** categorie di soggetti cui si applicano gli obblighi della disciplina NIS2.
- **Spear phishing:** messaggio fraudolento costruito su misura per un destinatario specifico.

- **STIX** (Structured Threat Information Expression): formato standard per rappresentare l'intelligence come oggetti collegati da relazioni.
- **Storage append-only**: archivio in cui i record si possono aggiungere ma non modificare o cancellare.
- **Super-timeline**: cronologia unificata degli eventi estratti da fonti diverse e normalizzati in un unico formato.
- **Tabletop**: esercitazione in cui un gruppo simula la gestione di un incidente attraverso la discussione di uno scenario.
- **Tasso di override**: frequenza con cui l'operatore corregge o respinge le proposte di un sistema automatico.
- **TAXII** (Trusted Automated Exchange of Intelligence Information): protocollo con cui un client invia oggetti STIX a una raccolta di un server o li richiede.
- **Test di regressione**: prova rieseguita dopo ogni modifica per verificare che il comportamento già validato di un sistema, compreso un sistema AI, non sia peggiorato e che gli errori già corretti non si ripresentino.
- **Timestomping**: alterazione intenzionale dei timestamp di un file.
- **TLP** (Traffic Light Protocol): marcatura che indica i limiti di ridistribuzione di un'informazione.
- **Triage forense**: ordinamento dei supporti e degli artefatti per priorità di esame, prima dell'analisi approfondita.
- **TTP** (Tattiche, tecniche e procedure): descrizione del modo in cui un avversario opera.
- **Tunnel vision**: tendenza a concentrare l'esame sugli elementi segnalati da uno strumento, trascurando quelli collocati in fondo alla coda di lavoro.
- **UEBA** (User and Entity Behavior Analytics): analisi del comportamento di utenti ed entità per individuare scostamenti dalla norma.
- **Unified Audit Log**: registro di audit di Microsoft 365 che raccoglie le attività di utenti e amministratori nei servizi del tenant.
- **Valore predittivo positivo**: probabilità che un elemento segnalato sia effettivamente positivo; coincide con la precision.
- **Verifica fuori banda**: conferma di una richiesta su un canale indipendente da quello da cui la richiesta proviene.
- **Vishing**: ingegneria sociale condotta tramite telefonate o messaggi vocali.
- **Volatility 3**: framework open source per l'analisi dei dump di memoria.
- **WAL** (write-ahead log): file in cui SQLite registra le pagine modificate prima di trasferirle nel database principale.
- **Write blocker**: dispositivo o software che impedisce le scritture sul supporto durante l'acquisizione.
- **YARA**: linguaggio per descrivere, con stringhe, sequenze di byte e condizioni logiche, le caratteristiche che identificano un malware o una famiglia.

- **Zeek:** sistema open source di monitoraggio della rete che produce metadati strutturati per ogni sessione.

Bibliografia e fonti

Elenco delle letture indicate nei capitoli, in ordine alfabetico.

- Anthropic, *Disrupting the first reported AI-orchestrated cyber espionage campaign*, novembre 2025, da leggere insieme alle osservazioni critiche pubblicate dopo la diffusione.
- B. Beyer, C. Jones, J. Petoff, N. R. Murphy (a cura di), *Site Reliability Engineering: How Google Runs Production Systems*, O'Reilly, 2016, capitoli 14 (*Managing Incidents*) e 15 (*Postmortem Culture: Learning from Failure*).
- B. Biggio, F. Roli, *Wild Patterns: Ten Years After the Rise of Adversarial Machine Learning*, Pattern Recognition, 2018.
- B. Carrier, *File System Forensic Analysis*, Addison-Wesley, 2005 (capitoli su NTFS).
- B. Strom et al., *MITRE ATT&CK: Design and Philosophy*, MITRE, 2018 (rev. 2020).
- C2PA, *C2PA Technical Specification* (Content Credentials).
- CISA, *Implementing Phishing-Resistant MFA*, 2022.
- Cisco Talos, «Olympic Destroyer Takes Aim At Winter Olympics», 12 febbraio 2018.
- Commissione europea, *Guidelines on prohibited artificial intelligence practices established by Regulation (EU) 2024/1689 (AI Act)*, febbraio 2025.
- Corte di giustizia dell'Unione europea, sentenza del 7 dicembre 2023, causa C-634/21 (SCHUFA Holding).
- D. Bianco, «The Pyramid of Pain», 2013 (rev. 2014).
- D. Brezinski, T. Killalea, *RFC 3227, Guidelines for Evidence Collection and Archiving*, 2002.
- D. Brezinski, T. Killalea, *RFC 3227, Guidelines for Evidence Collection and Archiving*, IETF, 2002.
- D. Chismon, M. Ruks, *Threat Intelligence: Collecting, Analysing, Evaluating*, MWR InfoSecurity, con il contributo di CPNI e CERT-UK, 2015.
- D. Plohmman, K. Yakdan, M. Klatt, J. Bader, E. Gerhards-Padilla, «A Comprehensive Measurement Study of Domain Generating Malware», *USENIX Security Symposium*, 2016.
- D. Quick, K.-K. R. Choo, «Impacts of increasing volume of digital forensic data: A survey and future research challenges», *Digital Investigation*, 11(4), 2014, pp. 273–294.
- D. Ucci, L. Aniello, R. Baldoni, «Survey of machine learning techniques for malware analysis», *Computers & Security*, 81, 2019.
- *Daubert v. Merrell Dow Pharmaceuticals, Inc.*, 509 U.S. 579 (1993); *Kumho Tire Co. v. Carmichael*, 526 U.S. 137 (1999).
- Direttiva (UE) 2022/2555 (NIS2) e D.Lgs. 4 settembre 2024, n. 138, in particolare artt. 25 e 38.
- Direttiva (UE) 2024/2853 sulla responsabilità per danno da prodotti difettosi.
- Documentazione di CAPEv2 e di YARA; MITRE ATT&CK, tecniche T1027 e T1497.
- Documentazione di Microsoft Security Copilot.

- Documentazione di Plaso (log2timeline) e di Timesketch.
- Documentazione ufficiale di TheHive e Cortex (StrangeBee) e di Shuffle.
- E. Casey, *Digital Evidence and Computer Crime*, 3^a ed., Academic Press, 2011, capitoli metodologici.
- ENFSI, *Best Practice Manual for the Forensic Examination of Digital Technology*, 2015.
- ENISA, *Good Practice Guide for Incident Management*, 2010.
- ENISA, *Threat Landscape*, edizione più recente.
- ENISA, *Threat Landscape*, edizioni annuali dal 2023, sezioni dedicate all'intelligenza artificiale.
- Europol Innovation Lab, *ChatGPT: The impact of Large Language Models on Law Enforcement*, 2023.
- F. Heiding, S. Lermen, A. Kao, B. Schneier, A. Vishwanath, *Evaluating Large Language Models' Capability to Launch Fully Automated Spear Phishing Campaigns: Validated on Human Subjects*, preprint arXiv:2412.00586, 2024; versione rivista: F. Heiding et al., *Evaluating Large Language Models' Ability to Automate Spear Phishing*, *Expert Systems with Applications*, 314, 2026.
- F. Pendlebury et al., «TESSERACT: Eliminating Experimental Bias in Malware Classification across Space and Time», *USENIX Security Symposium*, 2019.
- FireEye, *Highly Evasive Attacker Leverages SolarWinds Supply Chain to Compromise Multiple Global Victims With SUNBURST Backdoor*, dicembre 2020.
- FIRST, *Traffic Light Protocol (TLP) Version 2.0*, 2022.
- G. Engelen, V. Rimmer, W. Joosen, «Troubleshooting an Intrusion Detection Dataset: the CICIDS2017 Case Study», *IEEE Security and Privacy Workshops*, 2021.
- Google Threat Intelligence Group, *Adversarial Misuse of Generative AI*, gennaio 2025; *GTIG AI Threat Tracker: Advances in Threat Actor Usage of AI Tools*, novembre 2025; *GTIG AI Threat Tracker: Adversaries Leverage AI for Vulnerability Exploitation, Augmented Operations, and Initial Access*, maggio 2026.
- Gruppo di lavoro Articolo 29, *Linee guida sul processo decisionale automatizzato relativo alle persone fisiche e sulla profilazione* (WP251 rev.01), 2018, fatte proprie dall'EDPB.
- H. Aghakhani et al., «When Malware is Packin' Heat: Limits of Machine Learning Classifiers Based on Static Analysis Features», *NDSS Symposium*, 2020.
- H. Carvey, *Windows Forensic Analysis Toolkit*, 4^a ed., Syngress, 2014.
- H. Farid, *Photo Forensics*, MIT Press, 2016.
- H. S. Anderson, P. Roth, «EMBER: An Open Dataset for Training Static PE Malware Machine Learning Models», *arXiv*, 2018.
- I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016, capitoli 1 e 5.
- I. Sharafaldin, A. H. Lashkari, A. A. Ghorbani, «Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization», *ICISSP*, 2018.
- ISO/IEC 27035-1:2023, *Information security incident management, Part 1: Principles and process*.

- ISO/IEC 27037:2012, *Guidelines for identification, collection, acquisition and preservation of digital evidence*.
- ISO/IEC 27041:2015, *Guidance on assuring suitability and adequacy of incident investigative method*; ISO/IEC 27042:2015, *Guidelines for the analysis and interpretation of digital evidence*.
- ISO/IEC 27042:2015, *Guidelines for the analysis and interpretation of digital evidence*.
- ISO/IEC 27043:2015, *Incident investigation principles and processes*.
- J. Kornblum, «Identifying almost identical files using context triggered piecewise hashing», *Digital Investigation*, 3 (suppl.), 2006.
- K. Greshake et al., *Not What You've Signed Up For: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection*, 2023.
- Kaspersky GReAT, «The devil's in the Rich header», 8 marzo 2018.
- L. 20 maggio 1970, n. 300 (Statuto dei lavoratori), art. 4; L. 23 settembre 2025, n. 132, art. 11.
- L. Floridi et al., *AI4People. An Ethical Framework for a Good AI Society: Opportunities, Risks, Principles, and Recommendations*, Minds and Machines, 28, 2018.
- L. Zheng et al., *Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena*, NeurIPS 2023.
- Legge 18 marzo 2008, n. 48; codice di procedura penale, artt. 359 e 360.
- M. H. Ligh, A. Case, J. Levy, A. Walters, *The Art of Memory Forensics*, Wiley, 2014.
- M. Sikorski, A. Honig, *Practical Malware Analysis*, No Starch Press, 2012.
- Microsoft Threat Intelligence e OpenAI, *Staying ahead of threat actors in the age of AI*, febbraio 2024.
- Microsoft, documentazione di Microsoft Purview Audit e dello Unified Audit Log.
- MITRE, *ATLAS: Adversarial Threat Landscape for Artificial-Intelligence Systems*, atlas.mitre.org.
- MITRE, *ATT&CK Enterprise Matrix* e documentazione delle tecniche (attack.mitre.org).
- MITRE, *ATT&CK for Enterprise*, matrice e schede delle tecniche.
- MITRE, *D3FEND*, base di conoscenza delle contromisure difensive (d3fend.mitre.org).
- NATO, *AJP-2.1 Allied Joint Doctrine for Intelligence Procedures*, edizione corrente.
- NIST AI 100-1, *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*, 2023, e relativo *AI RMF Playbook*.
- NIST AI 100-1, *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*, 2023.
- NIST, *AI 100-2 E2025, Adversarial Machine Learning: A Taxonomy and Terminology of Attacks and Mitigations*, 2025.
- NIST, *AI 600-1, Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile*, 2024.
- NIST, *NISTIR 8006, NIST Cloud Computing Forensic Science Challenges*, 2020.
- NIST, *SP 800-61 Rev. 2, Computer Security Incident Handling Guide*, 2012 (ritirata il 3 aprile 2025 e sostituita dalla Rev. 3; utile per il modello a quattro fasi).

- NIST, *SP 800-61 Rev. 3, Incident Response Recommendations and Considerations for Cybersecurity Risk Management: A CSF 2.0 Community Profile*, 2025.
- NIST, *SP 800-84, Guide to Test, Training, and Exercise Programs for IT Plans and Capabilities*, 2006.
- NIST, *SP 800-86, Guide to Integrating Forensic Techniques into Incident Response*, 2006.
- OASIS, *CACAO Security Playbooks Version 2.0*, Committee Specification 01, 2023.
- OASIS, *STIX Version 2.1 e TAXII Version 2.1*, OASIS Standard, 2021.
- OWASP, *LLM AI Cybersecurity & Governance Checklist*, 2024.
- OWASP, *Top 10 for LLM Applications 2025*.
- OWASP, *Top 10 for LLM Applications*, edizione 2025.
- OWASP, *Top 10 for LLM Applications*, versione 2025.
- P. Kral, *Incident Handler's Handbook*, SANS Institute, 2012.
- P. Lewis et al., *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*, NeurIPS 2020.
- P. Lewis et al., «Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks», *Advances in Neural Information Processing Systems* 33, 2020.
- R. Ayers, S. Brothers, W. Jansen, *NIST SP 800-101 Rev. 1, Guidelines on Mobile Device Forensics*, 2014.
- R. Chesney, D. Citron, «Deep Fakes: A Looming Challenge for Privacy, Democracy, and National Security», *California Law Review*, 107, 2019.
- R. Jordaney et al., «Transcend: Detecting Concept Drift in Malware Classification Models», *USENIX Security Symposium*, 2017.
- R. Lyda, J. Hamrock, «Using Entropy Analysis to Find Encrypted and Packed Malware», *IEEE Security & Privacy*, 5(2), 2007.
- R. Parasuraman, D. H. Manzey, «Complacency and Bias in Human Use of Automation: An Attentional Integration», *Human Factors*, 52(3), 2010.
- R. Parasuraman, V. Riley, «Humans and Automation: Use, Misuse, Disuse, Abuse», *Human Factors*, 39(2), 1997.
- R. Sommer, V. Paxson, «Outside the Closed World: On Using Machine Learning for Network Intrusion Detection», *IEEE Symposium on Security and Privacy*, 2010.
- Reg. (UE) 2023/1543 e Dir. (UE) 2023/1544 (e-Evidence).
- Regolamento (UE) 2016/679 (GDPR), artt. 33–34; Direttiva (UE) 2022/2555 (NIS2); D.Lgs. 4 settembre 2024, n. 138.
- Regolamento (UE) 2016/679 (GDPR), in particolare artt. 5, 6, 22, 33–35 e capo V.
- Regolamento (UE) 2024/1689 (AI Act), art. 14, nel testo modificato dal Regolamento (UE) 2026/1744.
- Regolamento (UE) 2024/1689 (AI Act), artt. 12, 19 e 26, nel testo modificato dal Regolamento (UE) 2026/1744.
- Regolamento (UE) 2024/1689 (AI Act), in particolare artt. 3, 4, 5, 6, 12, 14, 25, 26, 50, 99 e allegato III, nel testo modificato dal Regolamento (UE) 2026/1744 (Omnibus digitale sull'IA).

- S. Axelsson, «The Base-Rate Fallacy and the Difficulty of Intrusion Detection», *ACM Transactions on Information and System Security*, 3(3), 2000.
- S. Es, J. James, L. Espinosa Anke, S. Schockaert, *RAGAs: Automated Evaluation of Retrieval Augmented Generation*, EACL 2024 (System Demonstrations), pp. 150–158; preprint arXiv:2309.15217, 2023.
- SigmaHQ, specifica del formato Sigma e repository delle regole.
- SigmaHQ, specifica Sigma e documentazione di pySigma e sigma-cli.
- T. Saito, M. Rehmsmeier, «The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets», *PLoS ONE*, 10(3), e0118432, 2015.
- U.S. Department of Justice, «Six Russian GRU Officers Charged in Connection with Worldwide Deployment of Destructive Malware and Other Disruptive Actions in Cyberspace», 19 ottobre 2020.
- U.S. Senate Committee on Commerce, Science and Transportation, *A «Kill Chain» Analysis of the 2013 Target Data Breach*, Majority Staff Report for Chairman Rockefeller, 26 marzo 2014.
- Un report pubblico recente a scelta (Mandiant, ESET o un advisory della CISA).
- V. Chandola, A. Banerjee, V. Kumar, «Anomaly Detection: A Survey», *ACM Computing Surveys*, 41(3), 2009.
- Volatility Foundation, documentazione di Volatility 3.
- X. Hu, J. Jang, M. Ph. Stoecklin, T. Wang, D. L. Schales, D. Kirat, J. R. Rao, «BAYWATCH: Robust Beaconing Detection to Identify Infected Hosts in Large-Scale Enterprise Networks», *IEEE/IFIP DSN*, 2016.