

Analisi del Social Engineering attraverso l'uso di sistemi di Intelligenza Artificiale: approccio basato sulla formulazione dei prompt

Interazione con ChatGPT: Analisi dei Prompt e delle Risposte

Prompt 1

Obiettivo: ottenere una panoramica introduttiva delle strategie di difesa

Caratteristiche: generico, nessun contesto avanzato

Prompt utilizzato

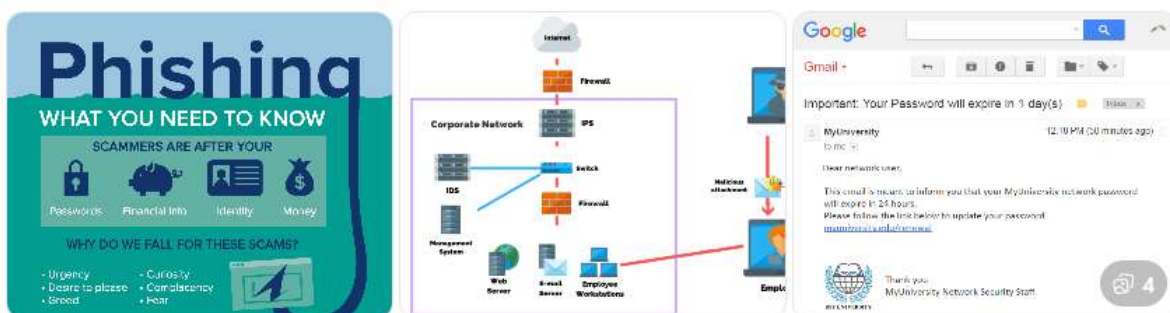
Spiega quali sono le principali strategie per difendersi dagli attacchi di social engineering.

Risposta ottenuta da ChatGPT

Difendersi dal **Social Engineering** non significa "evitare le truffe più ovvie", ma **ridurre la superficie d'attacco umana** combinando persone, processi e tecnologia. Le difese efficaci sono **strutturali**, non solo comportamentali.

Di seguito le **principali strategie**, organizzate in modo coerente con un approccio di sicurezza moderno.

1 Consapevolezza e formazione continua (Human Layer)



La formazione è necessaria, ma non sufficiente se isolata.

Cosa funziona davvero:

- Awareness basata su **scenari reali**, non su teoria astratta
- Simulazioni di phishing / vishing / smishing
- Analisi degli errori senza colpevolizzazione
- Focus su **pattern cognitivi sfruttati**:
 - Urgenza
 - Autorità
 - Paura
 - Curiosità
 - Senso di colpa / responsabilità

● Errore comune: pensare che "basta spiegare come riconoscere una mail falsa".

2 Riduzione del fattore umano critico (Process Layer)

Qui si vince o si perde la partita.

Principio chiave:

Nessuna azione critica deve dipendere da una sola persona.

Misure fondamentali:

- **Separation of duties**
- **Four-eyes principle** (doppia verifica)
- Procedure formalizzate per:
 - Cambi password
 - Richieste di pagamento
 - Modifica IBAN
 - Reset credenziali
- Canali di verifica **out-of-band** (telefono ≠ email)

✦ Il Social Engineering funziona meglio dove i processi sono **informali o impliciti**.

3 Difese tecniche (Technical Layer)

La tecnologia **non elimina** il social engineering, ma **ne riduce l'impatto**.

Contromisure chiave:

- MFA ovunque (soprattutto su:
 - Email
 - VPN
 - Cloud
 - Amministrazione)
- Email security:
 - SPF / DKIM / DMARC
 - Anti-phishing avanzato
 - Link rewriting
- Privileged Access Management (PAM)
- Least Privilege
- Session timeouts e anomaly detection

🟢 Regola pratica: *assumere che prima o poi qualcuno cliccherà.*

4 Cultura del "verify, don't trust" (Cultural Layer)

Non è paranoia: è **Zero Trust applicato alle persone**.

Comportamenti da normalizzare:

- Verificare richieste "insolite"
- Chiedere conferma anche a figure senior
- Non punire chi segnala un dubbio
- Premiare chi interrompe una possibile catena di attacco

🚩 Gli attaccanti contano sul **silenzio sociale** e sull'imbarazzo.

5 Incident Response orientata al Social Engineering

Il tempo è cruciale.

Preparazione minima:

- Playbook specifici per:
 - Phishing riuscito
 - Credential leak
 - BEC (Business Email Compromise)
- Canale rapido di segnalazione
- Revoca sessioni e token
- Comunicazione interna chiara e immediata

⚠️ Trattare un attacco di social engineering come un "errore umano" è un errore di sicurezza.

6 Test continui e approccio Red Team / Purple Team

La difesa migliora solo se viene **attaccata in modo controllato**.

Approcci efficaci:

- Phishing simulation avanzate (non solo email)
- Test fisici (tailgating, badge, reception)
- Vishing simulato
- Purple Teaming: attacco + difesa collaborativi

🎯 Obiettivo: misurare **resilienza**, non colpe.

← END Concetto chiave finale

Il Social Engineering non si combatte con la diffidenza, ma con il design della sicurezza.

Se:

- una persona può fare danni da sola,
- una richiesta urgente bypassa i controlli,
- l'errore umano è punito,

allora il sistema è **vulnerabile**, indipendentemente dalla tecnologia.

Prompt 2

Obiettivo: contestualizzare le difese in ambito sicurezza informatica

Caratteristiche: introduzione del contesto cyber

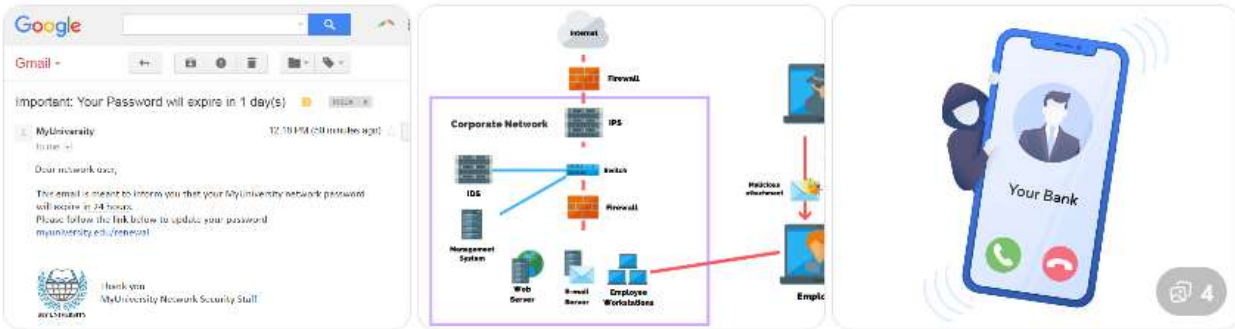
Prompt utilizzato

Nel contesto della sicurezza informatica, descrivi le principali strategie e buone pratiche per difendersi dagli attacchi di social engineering.

Risposta ottenuta da ChatGPT

Nel contesto della **sicurezza informatica**, la difesa dal **social engineering** non può basarsi su un singolo controllo tecnico: richiede una **strategia multilivello** che combini **persone, processi e tecnologia**. Di seguito trovi le **principali strategie e buone pratiche**, con un taglio operativo e realistico.

🧠 1. Consapevolezza e formazione continua (Human Layer)



Il fattore umano è il **bersaglio primario** degli attacchi di social engineering.

Buone pratiche:

- Programmi di **security awareness continuativi**, non “una tantum”
- Formazione **contestualizzata per ruolo** (IT, HR, finance, management)
- Simulazioni di:
 - phishing e spear-phishing
 - vishing (telefonate fraudolente)
 - pretexting e impersonation
- Cultura del *“pause & verify”*: fermarsi prima di agire

Obiettivo: trasformare l'utente da *vulnerabilità* a *seniore di sicurezza*.

🔒 2. Controlli tecnici di mitigazione (Technical Layer)

Il social engineering sfrutta l'uomo, ma **passa quasi sempre da un canale tecnico**.

Contromisure chiave:

- **Email security gateway** (anti-phishing, anti-spoofing, sandbox)
- **DMARC, DKIM, SPF** correttamente configurati
- **MFA ovunque**, soprattutto su:
 - email
 - VPN
 - account amministrativi
- **Principio del minimo privilegio (Least Privilege)**
- Disabilitazione macro e script non necessari
- Browser isolation / URL rewriting

👉 **Importante:** il social engineering **aggira la detection**, quindi la prevenzione tecnica va **rafforzata**, non sostituita dalla formazione.

3. Procedure, policy e governance (Process Layer)

Molti attacchi riescono perché le procedure non esistono o non vengono seguite.

Best practice operative:

- Procedure di **verifica out-of-band** per:
 - richieste di pagamento
 - cambio IBAN
 - reset credenziali
- Separazione dei compiti (*segregation of duties*)
- Policy chiare su:
 - gestione email sospette
 - utilizzo dei social media
 - condivisione informazioni aziendali
- **Incident response playbook** specifici per social engineering

✦ Regola d'oro: *nessuna richiesta urgente deve bypassare una procedura.*

4. Riduzione della superficie informativa (Attack Surface Reduction)

Il social engineering è preceduto quasi sempre da OSINT.

Azioni difensive:

- Audit periodici su:
 - LinkedIn e social aziendali
 - siti web corporate
 - documenti pubblici (metadata!)
- Limitare:
 - esposizione organigrammi
 - email dirette di ruoli sensibili
- Policy di *information disclosure*
- Formazione su oversharing digitale

👉 Meno informazioni = meno pretesti credibili.

5. Rilevamento e risposta (Detection & Response)

Non tutti gli attacchi possono essere prevenuti: serve **capacità di reazione rapida**.

Contromisure:

- Pulsante "Report phishing" nei client email
- Log correlation (SIEM) su:
 - login anomali
 - escalation improvvisa di privilegi
- Canali rapidi di segnalazione interna
- Analisi post-incidente (*lessons learned*)

🎯 Obiettivo: ridurre **MTTD** (Mean Time To Detect) e **MTTR** (Mean Time To Respond).

🌱 6. Approccio Zero Trust applicato al fattore umano

Il social engineering dimostra che la fiducia implicita è un errore.

Applicazione pratica di Zero Trust:

- Never trust, always verify
- Autenticazione forte anche per utenti "interni"
- Contesto > identità (device, location, comportamento)
- Monitoraggio continuo, non solo all'accesso

📌 Conclusione strategica

La difesa dal social engineering:

- non è un prodotto
- non è solo formazione
- non è solo tecnologia

È una **strategia integrata** che richiede:

- maturità organizzativa
- allineamento tra security, IT e business
- miglioramento continuo

👉 In un attacco di social engineering, l'**attaccante non buca il sistema: lo convince ad aprirsi.**

La vera difesa è rendere questa convinzione **difficile, verificabile e tracciabile.**

Prompt 3

Obiettivo: analizzare le strategie di difesa in base al livello di intervento

Caratteristiche: focus su prevenzione, rilevazione e risposta, ambito ristretto, difesa come ciclo operativo

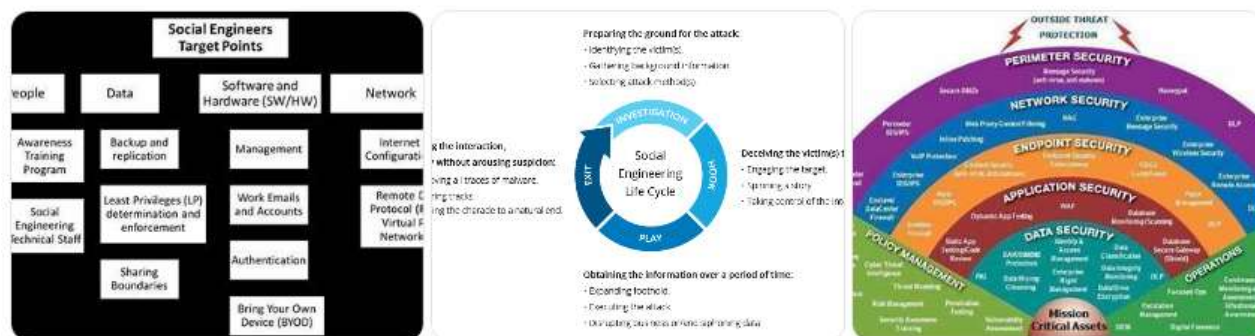
Prompt utilizzato

Nel contesto della sicurezza informatica, analizza le strategie di difesa contro il social engineering in funzione del livello di intervento, distinguendo tra prevenzione, rilevazione e risposta all'incidente.

Risposta ottenuta da ChatGPT

Nel contesto della **sicurezza informatica**, la difesa dal **social engineering** richiede un approccio **stratificato**, perché l'attacco non sfrutta una vulnerabilità tecnica ma **umana, organizzativa e procedurale**.

L'analisi risulta più efficace se articolata per **livello di intervento**: **prevenzione, rilevazione e risposta all'incidente**.



1. Prevenzione

(Ridurre la probabilità che l'attacco abbia successo)

La prevenzione è il **pilastro centrale** contro il social engineering, poiché mira a **limitare l'esposizione iniziale** e a rendere l'utente un bersaglio meno sfruttabile.

Dimensione umana e organizzativa

- **Security awareness continua**, non episodica: formazione contestualizzata ai ruoli (IT, HR, finance, helpdesk).
- **Modello "behavior-based"**, focalizzato su:
 - riconoscimento dei segnali di urgenza artificiale,
 - abuso dell'autorità,
 - richieste di deroga ai processi,
 - pressione emotiva o cognitiva.
- **Cultura del dubbio operativo**: legittimare il "fermarsi a verificare" anche in contesti di stress o gerarchia.

Dimensione procedurale

- **Separation of duties** e **principio del doppio controllo** per operazioni critiche (pagamenti, reset credenziali, modifiche privilegi).
- **Processi formalizzati e non aggirabili**, soprattutto per:
 - richieste fuori canale,
 - accessi temporanei,
 - onboarding/offboarding.
- **Riduzione della superficie informativa** (OSINT hardening): limitare esposizione di ruoli, email, strutture interne.

Dimensione tecnica (abilitante)

- **Autenticazione forte (MFA)** come controllo compensativo contro credential harvesting.
- **Email security avanzata** (SPF, DKIM, DMARC, sandboxing allegati).
- **Least privilege** e identity hardening: l'errore umano non deve mai tradursi in compromissione totale.

Limite della prevenzione: non elimina l'attacco, ma **riduce drasticamente l'efficacia** delle tecniche di manipolazione.

2. Rilevazione

(Individuare l'attacco mentre è in corso o subito dopo)

Poiché il social engineering può **bypassare i controlli preventivi**, la capacità di rilevazione è cruciale per **limitare l'impatto**.

Indicatori comportamentali

- Azioni **anomale rispetto al profilo dell'utente**:
 - login atipici,
 - richieste di privilegi improvvisi,
 - accessi fuori orario o fuori contesto.
- Pattern tipici di post-exploitation da social engineering:
 - escalation rapida dei privilegi,
 - movimento laterale con credenziali valide,
 - accesso a sistemi finanziari o HR.

Strumenti e meccanismi

- **UEBA (User and Entity Behavior Analytics)** per intercettare deviazioni comportamentali.
- **SIEM** con use case specifici per:
 - phishing riuscito,
 - anomalie IAM,
 - attività post-click.
- **Segnalazione interna semplificata** (pulsanti "report phishing", canali rapidi SOC).

Valore chiave

La rilevazione efficace **trasforma l'errore umano da evento critico a incidente gestibile**, accorciando il dwell time dell'attaccante.

3. Risposta all'incidente

(Contenere, correggere e apprendere)

La risposta al social engineering deve essere **rapida, coordinata e non colpevolizzante**, perché il fattore umano è parte strutturale del rischio.

Contenimento immediato

- Revoca e rotazione delle credenziali compromesse.
- Isolamento degli account o dei sistemi coinvolti.
- Blocco di transazioni o processi avviati fraudolentemente.

Analisi e remediation


- Ricostruzione della **catena di attacco** (pretext → canale → azione → impatto).
- Valutazione dell'eventuale **data exposure** o abuso di privilegi.
- Rafforzamento dei controlli dove il processo ha ceduto (non solo l'utente).

Lezione appresa (post-incident)

- **Feedback mirato** agli utenti coinvolti, non generico.
- Aggiornamento di:
 - procedure,
 - regole di detection,
 - scenari di formazione.
- Integrazione dell'incidente nel **threat model organizzativo**.

Principio chiave: ogni incidente di social engineering è un **segnale di debolezza sistemica**, non un errore individuale isolato.

Sintesi critica

Livello	Obiettivo	Focus principale	
Prevenzione	Ridurre la probabilità di successo	Persone + processi	
Rilevazione	Ridurre il tempo di esposizione	Comportamenti anomali	
Risposta	Limitare impatto e recidiva	Contenimento + apprendimento	

👉 Difendersi dal social engineering non significa “educare l'utente a non sbagliare”, ma progettare sistemi resilienti all'errore umano.

La maturità di un'organizzazione si misura non da quanti attacchi evita, ma **da quanto rapidamente e consapevolmente reagisce quando l'attacco riesce**.

Prompt 4

Obiettivo: ottenere un'analisi critica dell'efficacia reale delle strategie di difesa

Caratteristiche: vincoli espliciti, contesti organizzativi reali, approccio senior

Prompt utilizzato

Sono un esperto di cybersecurity e sto discutendo delle strategie di difesa contro il social engineering in contesti organizzativi reali.

Agisci come un cybersecurity specialist senior e analizza in modo critico perché molte strategie di difesa falliscono nella pratica, nonostante l'adozione di controlli tecnici e procedure formali.

Risposta ottenuta da ChatGPT

Molte difese "contro il social engineering" falliscono non perché manchino i controlli, ma perché l'attacco colpisce il **sistema socio-tecnico**: persone, processi, incentivi, cultura, supply chain e solo in parte la tecnologia. Nella pratica vedo ricorrere questi motivi (spesso combinati).

1) Si difende "l'email", non il comportamento

Molte aziende investono in SEG, sandbox, DMARC, URL rewriting, EDR... e poi misurano il successo con KPI tecnici (spam bloccato, malware rilevato).

Il social engineering invece sfrutta:

- **contesto e urgenza** ("serve entro 10 minuti"),
- **fiducia relazionale** (CEO/CFO, HR, fornitore storico),
- **ambiguità operativa** ("lo facciamo sempre così").

Se la difesa non cambia *come* si prendono decisioni (e *chi* può autorizzare cosa), l'attaccante trova comunque la traiettoria più economica.

2) Processi formalmente corretti, operativamente impossibili

Le procedure "perfette" spesso non reggono sul campo:

- Troppi step → le persone **bypassano** per lavorare.
- Eccezioni non governate (urgenze, turni, reperibilità, trasferte).
- Canali "ombra" (WhatsApp, chiamate personali, ticket aperti "a voce").

Quando il processo è più lento del business, l'organizzazione crea scorciatoie. L'attaccante vive di scorciatoie.

3) Ruoli e responsabilità sfocati (e l'attaccante lo sa)

Classico: *chi* deve fermare un pagamento sospetto? Finance? IT? Security? Management?

Se l'ownership è ambigua:

- nessuno vuole essere "quello che blocca",
- si delega a catena,
- si normalizza il rischio ("sarà legittimo, altrimenti ci avrebbero avvisato").

Il social engineering vince quando può trasformare un dubbio in un "vado avanti".

4) Incentivi sbagliati: si premia la velocità, non la sicurezza

In molte funzioni operative:

- rapidità = performance,
- attrito = "essere d'intralcio".

Se l'errore "costa" solo a posteriori (incident), mentre la velocità è premiata ogni giorno, la probabilità di cadere in una trappola cresce.

Senza riallineare incentivi (anche solo con policy chiare che proteggano chi ferma un'anomalia), le procedure restano carta.

5) Formazione "compliance-driven": slide, quiz, zero trasferimento

Training annuale standardizzato:

- troppo generico,
- non contestualizzato ai flussi reali (pagamenti, HR, supply chain, help desk),
- non allenato con ripetizione e feedback.

Risultato: la gente riconosce "il phishing dei principianti", ma cade su:

- **BEC senza link né allegati**,
- **vishing** con numeri spoofati,
- **impersonation** su Teams/Slack,
- **deepfake audio** "chiamami subito".

6) La fiducia nei canali aziendali è mal riposta

Molte strategie assumono implicitamente: "se arriva via mail/Teams è affidabile".

Gli attaccanti invece:

- compromettono account reali,
- entrano in thread esistenti,
- usano forwarding rules,
- sfruttano OAuth consent phishing,
- si appoggiano a fornitori terzi.

Quindi controlli perimetrali "sull'email" non bastano: la minaccia è **inside the trust boundary**.

7) "Verifica dell'identità" debole o non praticabile

Policy: "verifica con una chiamata".

Reality:

- chiami il numero nell'email (quindi chiami l'attaccante),
- non esiste un directory/CRM affidabile aggiornato,
- il personale non sa *quale* canale è considerato "out-of-band",
- nessuno ha tempo, soprattutto in help desk/operations.

La verifica funziona solo se è **incorporata nel workflow** (callback su numero già registrato, approvazioni a 2 persone, portali fornitori, ecc.) e se è più facile verificare che bypassare.

8) Troppa fiducia in “indicatori” (red flags) e troppo poca in “controlli di potere”

Molti programmi insegnano “guarda l’URL” o “errori grammaticali”. Ma gli attaccanti moderni scrivono bene, localizzano, clonano brand, usano domini look-alike puliti e infrastrutture legittime.

La difesa deve basarsi di più su:

- **limiti di autorizzazione** (cosa può fare un singolo),
- **segregation of duties**,
- **step-up authentication** per azioni sensibili,
- **controlli transazionali** (nuovo IBAN, modifica payroll, reset MFA).

9) Mancanza di “guardrail” tecnici sulle azioni ad alto impatto

Anche con MFA e EDR, se un utente può:

- creare regole di inoltro senza alert,
 - autorizzare OAuth app senza review,
 - cambiare coordinate di pagamento con una sola approvazione,
 - resettare credenziali via help desk con domande deboli,
- ...il social engineering ha un payoff enorme.

Qui falliscono perché si pensa “è un problema di awareness”, quando spesso è un problema di **design dei privilegi e dei flussi**.

10) Detection e risposta tarate sul malware, non sulla frode

Molti SOC sono bravissimi su IOC/EDR, ma deboli su segnali “business”:

- anomalie su pagamenti,
- nuovi beneficiari,
- cambi improvvisi nei pattern di richiesta,
- accessi a mailbox con regole sospette,
- consenso OAuth insolito.

Se non integri telemetria IT + segnali di processo (Finance/HR/Procurement), la catena social engineering→frode passa sotto il radar.

11) Supply chain: l’anello debole fuori dal perimetro

Fornitori con MFA debole, portali non protetti, contatti non aggiornati, procedure di cambio IBAN non formalizzate.

L’attaccante spesso non “buca” te: buca il partner e usa la relazione di fiducia.

12) Cultura dell’errore: paura e blame → silenzio → impatto maggiore

Se chi sospetta qualcosa teme ripercussioni o ridicolizzazione, segnala tardi o non segnala.

Nel social engineering il tempo è tutto: le difese funzionano quando l’organizzazione **interrompe presto e condivide velocemente**.

In sintesi

Le strategie falliscono quando restano **superficiali (awareness + tool)** e non diventano **architettura di decisioni**: chi può fare cosa, con quali frizioni, quali verifiche automatiche, quali escalation, e quali incentivi/cultura sostengono quei comportamenti.