Contents

[3.4] Bypass di Fiducia Basato sul Gradimento

1. Definizione Operativa: La vulnerabilità in cui un rapporto positivo, una similarità percepita o un gradimento personale per un richiedente causa a un individuo di aggirare i controlli di sicurezza o le procedure che normalmente seguirebbe.

2. Metrica Principale e Algoritmo:

- **Metrica: Tasso di Bypass per Affinità (ABR)**. Formula: `ABR = N_bypass / N_richieste`, dove `N_richieste` è il numero di richieste da un'entità "gradita" (ad esempio, un collaboratore frequente) e `N_bypass` è quante di quelle hanno portato a un bypass della policy.
- **Pseudocodice:**

  python

```
def calculate_abr(access_logs, chat_logs, hr_data):
    """
    hr_data: Dati dal sistema HR per mappare la struttura organizzativa.
    """
    abr_results = {}
    # 1. Identifica le coppie "ad alta affinità": utenti che interagiscono frequentemente
    affinity_pairs = find_high_affinity_pairs(chat_logs, hr_data, min_interactions=10)

    for user_a, user_b in affinity_pairs:
        # 2. Ottieni le richieste da user_b a user_a
        requests = get_requests_from_user(chat_logs, user_b, user_a)
        bypass_count = 0
        for req in requests:
            # 3. Verifica se la richiesta ha portato a un bypass della sicurezza (ad esemp
            if led_to_bypass(access_logs, user_a, req):
                bypass_count += 1

        total_requests = len(requests)
        ABR = bypass_count / total_requests if total_requests > 0 else 0
        abr_results[(user_a, user_b)] = ABR
    return abr_results
```

- **Soglia di Allerta:** `ABR > 0.3` (Oltre il 30% delle richieste da un collega ad alta affinità risulta in un bypass).

3. Fonti di Dati Digitali (Input dell'Algoritmo):

- **API Piattaforma di Comunicazione (Slack/Teams):** Per misurare la frequenza di interazione e il contenuto tra gli utenti. Campi: `user_from`, `user_to`, `message_count`, `sentiment_score` (se disponibile).

- **API Database HR:** Per comprendere le strutture gerarchiche formali e le appartenenze ai team.
- **Log di Accesso (SharePoint, GitHub, File Server):** Per verificare se le azioni di condivisione erano conformi alla policy (ad esempio, è stato referenziato un ticket di approvazione?). Campi: `user`, `action`, `target`, `timestamp`.

4. Protocollo di Audit Umano-Umano: Esamina i ticket di approvazione dell'accesso. Per un campione di approvazioni, intervista chi ha concesso l'accesso: "Hai approvato la richiesta X per [Collega Y]. Puoi descrivermi il tuo processo di pensiero? Il fatto che conosci bene e lavori a stretto contatto con loro ha reso l'approvazione più facile o veloce rispetto a se provenisse da uno sconosciuto?".

5. Azioni di Mitigazione Consigliate:

- **Mitigazione Tecnica/Digitale:** Implementa campi obbligatori nei flussi di lavoro per le richieste di accesso che forzano chi concede l'accesso a citare la giustificazione aziendale e la regola di policy che si applica, indipendentemente dal richiedente.
- **Mitigazione Umana/Organizzativa:** Formazione con esercitazioni di ruolo dove un collega ben gradito effettua una richiesta inappropriata, insegnando al personale come dire "no" con eleganza mantenendo il rapporto.
- **Mitigazione del Processo:** Introduci un principio di "quattro occhi" per certeSe azioni ad alto rischio richieste da individui al di fuori di un flusso di lavoro formale predefinito (come un sistema di ticketing).