TOPIC : DEVELOPMENT TEAM ACTION ITEMS

1. Severity Normalization Logic

- **Discussion:** Cloud providers (AWS, Azure, GCP) use non-standardized risk scales. Such as, AWS uses '7.0 - 8.9' score, Azure uses 'High'
- **Action Item:** Define a backend **Translation Layer** to map all provider-specific severities into our dashboard's four buckets (Critical, High, Medium, Low). We must agree on the threshold for what qualifies as Critical to ensure the Global Toggle is accurate.

2. Tagging Compliance & Fallback Strategy

- **Discussion:** The 'Risk Leaderboard' depends on cloud resources having metadata tags.
- **Action Item:** We need to handle **'The Orphaned Resource'** problem. If a resource lacks an owner tag, should the backend assign it to a default Global Infrastructure team, or can we programmatically lookup the IAM identity that created the resource?

3. API Aggregation & Performance

- **Discussion:** Fetching real-time data from 100+ accounts simultaneously can trigger API rate limits (throttling) and slow down the UI.
- **Action Item:** Implement a **Caching Strategy**. We should determine which metrics (like the 'Global Health Score') can be cached for 30 minutes, and which (like 'New Critical Alerts') require a real-time event-driven push using AWS EventBridge or similar services.

4. Deep-Link & Payload Integration

- **Discussion:** The 'Assign to Owner' button needs to provide developers with enough context to act immediately without navigating back to our dashboard.
- **Action Item:** Define the **Webhook Payload** for Jira and Slack. This must include the unique Resource ID, a direct link to the Cloud Console, and the 'Remediation Snippet' so the developer can patch the issue directly from their ticket.