**🧪 Lab: Threat Modeling with OWASP Threat Dragon (Level 0 and Level 1 DFD)**

**🎯 Objective:**

Students will learn how to create Level 0 and Level 1 DFDs using Threat Dragon and identify possible threats in a microservices-based cloud architecture (EKS with API Gateway and App Mesh).

**🔧 Pre-requisites:**

* Basic understanding of DFDs (Processes, Data Stores, External Entities, Data Flows)
* A browser with internet access
* Threat Dragon Web App: https://owasp.org/www-project-threat-dragon/

**🏗 Reference Architecture (Used in Lab):**

* **Title:** Expose Microservices Using Amazon EKS
* **URL:** [AWS Architecture PDF](https://d1.awsstatic.com/architecture-diagrams/ArchitectureDiagrams/expose-microservices-using-eks-ra.pdf)
* **Components involved:**
  + Clients (external users)
  + Amazon Route 53
  + Amazon API Gateway
  + Amazon EKS (Kubernetes)
  + App Mesh
  + Microservices (front-end, internal services)
  + Amazon CloudWatch
  + AWS WAF

**✅ Step-by-Step Instructions:**

**🧩 Step 1: Launch Threat Dragon**

1. Open browser and go to Threat Dragon Web App
2. Click **“Use Threat Dragon Web”**
3. Create a new model:
   * Model name: EKS Threat Model
   * Version: 1.0
   * Description: Threat model for EKS microservices exposure architecture
   * Click **“Create model”**

**🌀 Step 2: Create Level 0 DFD (High-Level Overview)**

**Goal:** Show main external actors, API gateway, and EKS cluster as one single processing block.

**Diagram Elements:**

| **Element** | **Label** |
| --- | --- |
| External Entity | Client |
| Process | API Gateway & EKS |
| External Entity | Monitoring (CloudWatch) |
| External Entity | DNS (Route 53) |
| External Entity | WAF |

**Actions:**

1. Add External Entity → Label as **Client**
2. Add External Entity → Label as **Route 53**
3. Add External Entity → Label as **WAF**
4. Add Process → Label as **API Gateway & EKS Services**
5. Add External Entity → Label as **Monitoring Tools**
6. Add Data Flows:
   * Client → Route 53: DNS Request
   * Client → WAF: HTTPS Request
   * WAF → API Gateway & EKS: Filtered Request
   * API Gateway & EKS → Monitoring: Logs/Alerts

📌 **Save this as Level 0** in your model notes.

**🧪 Step 3: Add Threats to Level 0**

Click on any element, and under the "Threats" tab, explore possible threats. Some examples:

* **Client → WAF:** *Spoofing or Injection*
* **WAF → API Gateway & EKS:** *Bypassing filtering rules*
* **API Gateway & EKS → Monitoring:** *Data Leakage, Information Disclosure*

**🔍 Step 4: Create Level 1 DFD (Drill Down into EKS)**

**Goal:** Show internal architecture – Microservices, App Mesh, and API Gateway inside EKS.

**Diagram Elements:**

| **Element** | **Label** |
| --- | --- |
| External Entity | WAF |
| Process | API Gateway |
| Process | Frontend Service (Microservice) |
| Process | Internal Microservices (x2) |
| Data Store | App Mesh |
| External Entity | CloudWatch |

**Actions:**

1. Create a new diagram or layer in same model
2. Add WAF → API Gateway
3. API Gateway → Frontend Service
4. Frontend Service → Internal Microservice A
5. Internal Microservice A ↔ Microservice B (via App Mesh)
6. All services send logs to CloudWatch

**Data flows:**

* WAF → API Gateway: HTTPS Request
* API Gateway → Frontend: REST Call
* Frontend → Microservice A: Service Call
* Microservice A ↔ Microservice B: gRPC over App Mesh
* Services → CloudWatch: Logs, Metrics

**🔐 Step 5: Add Threats to Level 1 DFD**

Sample threats to add:

| **Flow or Component** | **Sample Threat** |
| --- | --- |
| API Gateway → Frontend | Injection, Broken Auth |
| Microservices ↔ App Mesh | Man-in-the-middle, TLS misconfig |
| Services → CloudWatch | Information disclosure |

Use built-in Threat Dragon threat list or add custom ones.

**📤 Step 6: Export or Save the Model**

* Go to **Menu > Export**
* Save as .json (Threat Dragon format) or .png (image)
* You can also print for submission

**📘 Lab Deliverables (for Students)**

1. Screenshot of Level 0 and Level 1 DFDs
2. Exported .json file from Threat Dragon
3. Brief written summary:
   * Threats identified at each level
   * Recommendations or mitigations for top 3 risks

**🧠 Extension Activity (Optional)**

Have students classify threats using **STRIDE**:

* **S**poofing
* **T**ampering
* **R**epudiation
* **I**nformation Disclosure
* **D**enial of Service
* **E**levation of Privilege