Threat Modeling

Threat modeling and analysis provides a complete view about the security of a system. It is performed using a systematic and strategic way for identifying and enumerating threats to a system.

1. Some Common Definition (RFC 2828)

Vulnerability:
" A flaw or weakness in a system's design, implementation, or operation and management that could be exploited to violate the system's security policy"

Threat:
"A potential for violation of security, which exists when there is a circumstance, capability, action, or event that could breach security and cause harm... a threat is a possible danger that might exploit a vulnerability"

Attack:
"An assault on system security that derives from an intelligent threat, to evade security services and violate the security policy of a system."

2. Modeling Phases

We are currently following an iterative step, starting from the higher level – identifying major components, assets, and identification of threats from overall perspective. In the second part, we are performing threat analysis for each of the earlier identified components.. Finally, we merge the analysis report to one.

Questions: component breakdown or use case breakdown is better option ?

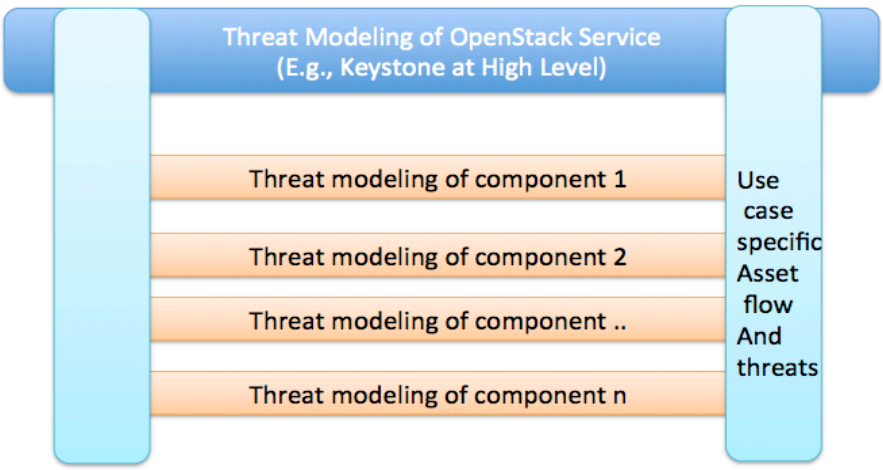
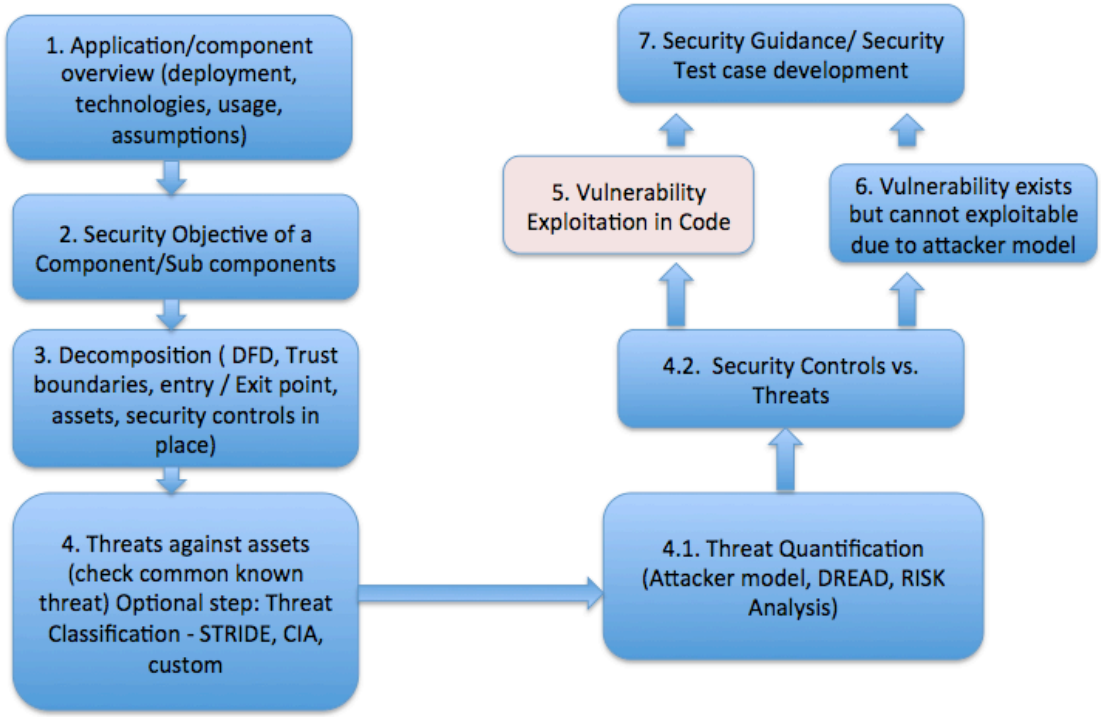


Figure 1: An iterative process for threat identification

Currently, we follow simplified steps for modeling (Do not try to reinvent the wheel here – Adjusted the OWSAP steps)



Figures 2: Simplified steps for threat modeling

Steps:

1. Application/component overview (deployment, technologies, usage, assumptions)

2. Security Objective of a Component/Sub components

3. Decomposition (Data Flow Diagrams - DFD, Trust boundaries, entry/exit point, assets, security controls in place)

4. Threats against assets (Derive known threats). STRIDE can be used for threat identification and classification.

4.1. Threat Quantification (Attacker model, DREAD, RISK Analysis)

4.2 Security Controls vs. Threats.

5. Vulnerability Exploitation in Code

6. Vulnerability exists but cannot be exploitable due to attacker model

7. Security Guidance/ Security Test case developments

3. Related Assumptions for the Threat Modeling:

Threat Agents:

In broad, we consider three kinds of threat agent (Attackers).

ID	Name	Details
IA-U	Internet Attacker– Unauthorized	
IA-A	Internet Attacker – Authorized	
IN-I	Internal Attacker - Insider	

Threat Categorization

We can follow STRIDE model (to understand STRIDE model, look at OWSAP documents)

ID	**Details**
SPOOFING	
TAMPERING	
REPUDIATION	
INFORMATION DISCLOSURE	
DEINIAL OF SERVICE	
ELEVATION OF PRIVILEDGE	

Threat Agent Capabilities

Level	Name
0	Script Kiddies
1	Motivated individuals
2	Highly Capable Individuals
3	Serious Organized crime
4	Intelligence services

Likelihood of a Threat

Level	Definition
Low	
Medium	

High	

Threat DB:

A set of known threats against which we perform threat analysis. The list is not exhaustive but a good starting point. Also, it is a time consuming tasks to check each and every threats and attack vectors for each asset. Details in Threat_DB_Ref.xls

4. Identification of Threats

In final analysis, we follow a simplified process as exemplified in figure 3. In this picture, top row elements are generated from the step 1 to 3 from figure 2. These are dynamic parameters. The bottom row is the result. The middle rows are static elements – defined in section 3.

Based on this information, we identify threats for each asset accessed or modified by the proxies (operations) in the data flows diagram.

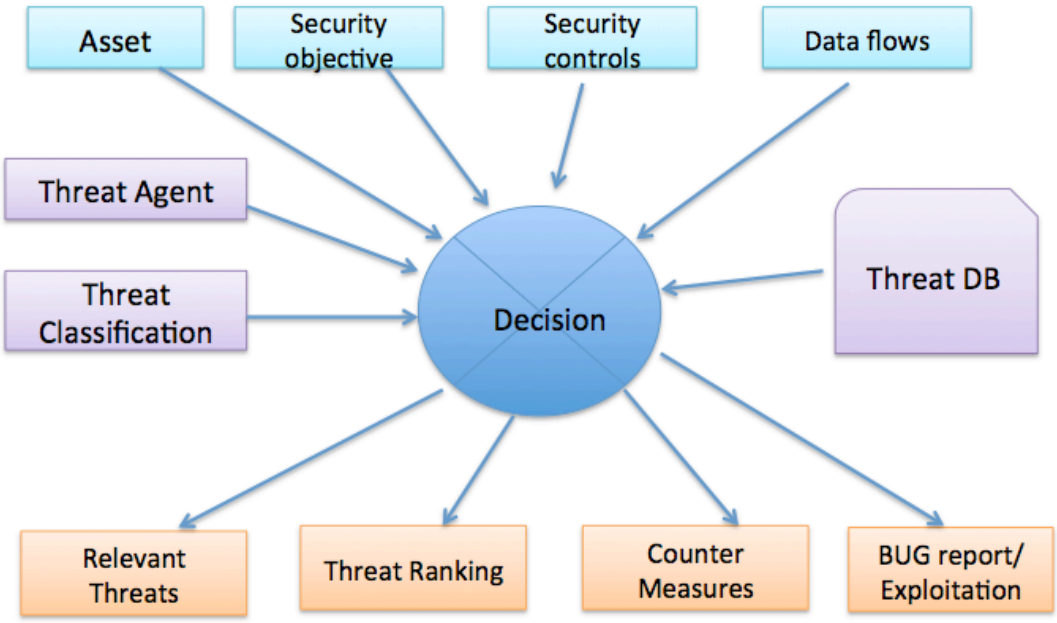


Figure 3: Threat Analysis - simplified

In other words,

```
# Logic to find related threats

for each asset_A in Asset:
   for each operations in DataFlows:
       modified_Asset_A = operation(asset_A)
```

```
for each threat_group in STRIDE:
      attack_vectors = Find attack_vectors related to
      threat_group from Threat_DB

      possible_Threat  = Threats exploitable  by using
      attack_vectors on asset_A to perform an operation

     ranked_Threat = Ranking(possible_Threat)
     return ranked_Threat
```