# SAPM Meeting
# Security

# What QAs are Important

- **Confidentiality:** Only those who should have access are given access.
- **Integrity:** Data or services are not subject to unauthorised manipulation
- **Availability:** the system is available for legitimate use.
- **Mechanisms that support CIA:** Authentication, nonrepudiation, authorisation.

# General Security Scenario

- **Source:** Humans or systems that may or may not have been identified and can be either inside or outside the organisation
- **Stimulus:** Unauthorized attempt to access, manipulate, or disable the artifact.
- **Artifact:** System services, data, components, data produced or consumed by the system.
- **Environment:** online, offline, connected to network, disconnected to network, behind firewall, fully/partially operating, not operational

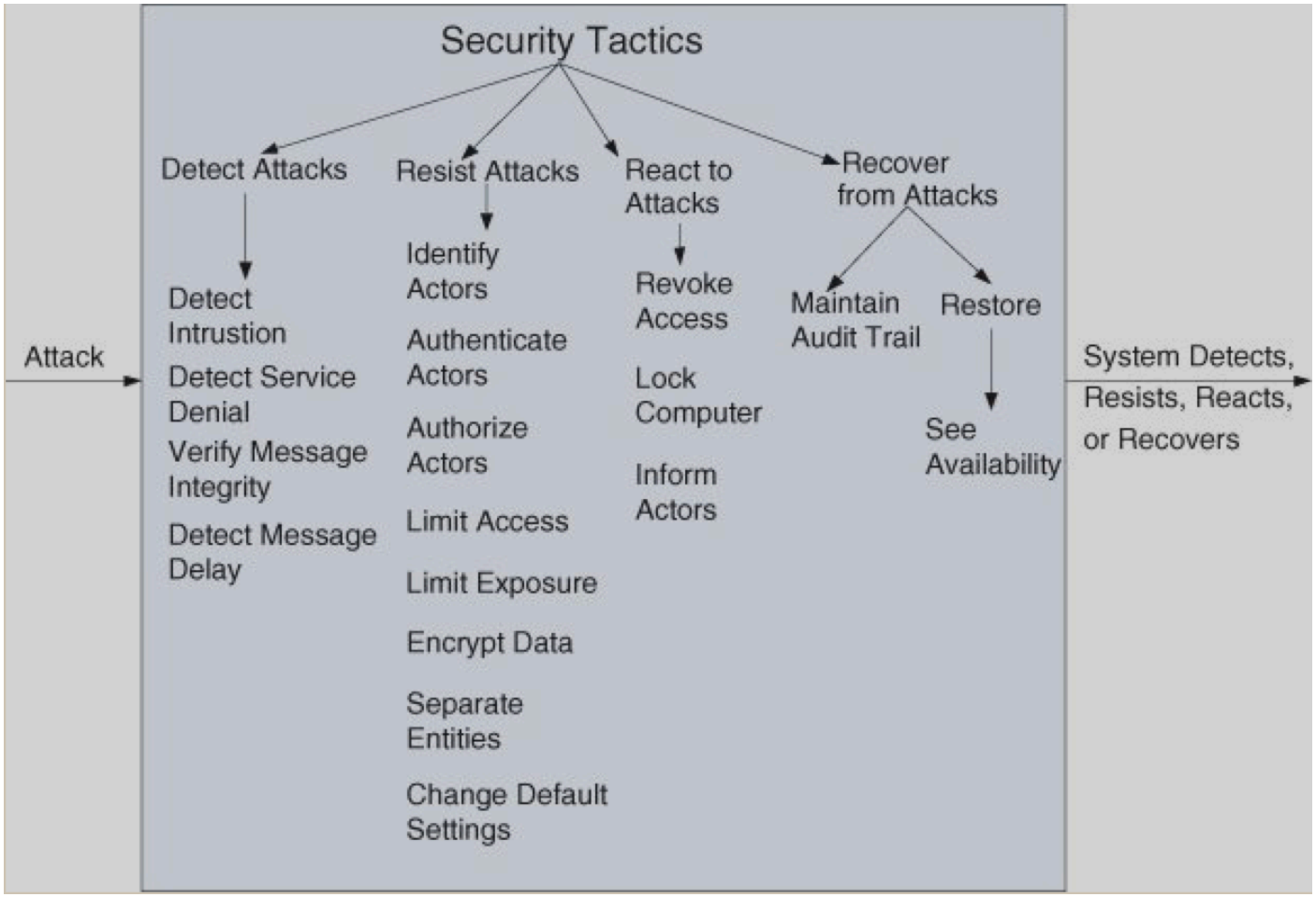
# Generic Security Scenario

- **Response:** Transactions are carried out so that:
  - Data or services are protected from unauthorized access
  - Data or services are not manipulated without authorization
  - Parties in a transaction are identified with assurance
  - Parties cannot repudiate their participation
  - Data, resources etc are available for legitimate use
  - System records: access, modification of data, resources or services.
  - Appropriate people are notified when threat is identified.
- **Response Measure:** assessment of the degree of compromise, temporal and spatial data on the compromise, how many attacks were resisted, how much data is vulnerable

# Concrete Security Scenario: Denial of Service

- **Source:** A wide range of systems with different IP addresses
- **Stimulus:** Accesses to the service provided
- **Artifact:** The service we are concerned with.
- **Environment:** Normal operation
- **Response:** Detect abnormal load
- **Response Measure:** Mode of operation is changed to ensure normal service to trusted IP addresses.

# Tactics


Security Tactics
Detect Attacks
Resist Attacks
React to Attacks
Recover from Attacks
Detect Intrustion
Detect Service Denial
Verify Message Integrity
Detect Message Delay
Identify Actors
Authenticate Actors
Authorize Actors
Limit Access
Limit Exposure
Encrypt Data
Separate Entities
Change Default Settings
Revoke Access
Lock Computer
Inform Actors
Maintain Audit Trail
Restore
See Availability
Attack
System Detects, Resists, Reacts, or Recovers

# Concrete Scenario

- Consider a service with no protection.
- This would fail our scenario.
- If we require the system to maintain service to key users. We require:
  - Some means of detecting anomalous service demand.
  - Service now operates in two modes: normal mode and priority users mode
  - We add a layer to the architecture that filters out non-priority user service demands in priority users mode.

# Allocation of Responsibilities

- For all security-sensitive system responsibilities, do the following:
  - Ensure all actors have identities
  - Authenticate identities
  - Check authorizations
  - Ensure authorization is required for all such actors
  - Log attempts, successes and failures on all sensitive operations
  - Ensure data is encrypted
  - Ensure responsibilities are allocated to appropriate actors.

# Coordination Model

- Ensure coordination mechanisms use authentication and authorisation.
- Ensure coordination mechanisms are not vulnerable to impersonation, tampering, interception, …
- Ensure data involved in coordination is protected using encryption.
- Monitor level of demand for communication to identify excessive demands

# Data Model

- Ensure there is a valid data model that disallows invalid data flows.
- Ensure logging of access, modification and attempted access or modification.
- Data is protected in flight and at rest using appropriate encryption.
- Ensure appropriate backup/recovery mechanisms are in place.

# Mapping among Architectural Elements

- Explore how different mappings change the way users can access resources.
- Ensure for all of these mappings the models of access and authorisation are preserved.
  - Actors should be identified and authenticated
  - Use appropriate authorisation mechanisms
  - Ensure logging is enabled
  - Ensure data is protected by encryption
  - Recognise impact of attack on resources
- Ensure recovery from attack is possible

# Resource Management

- Explore the overheads resulting from monitoring, detecting, preventing and recovering from attacks.
- Analyse how a user can access and make demands on critical resources.
- Manage resource access to ensure malicious use of resource is detected and managed.
- Identify the potential for corruption/contamination and how to manage this.
- Explore the potential for resource use to be used as a covert channel to transmit data.
- Limit resources used to manage attempts at unauthorised use.

# Binding Time

- Explore the consequences of varying binding times on the ability to trust an actor or component.
- Put in place appropriate mechanisms to ensure trust given binding time.
- Explore potential impact on resource use, capacity/throughput, response time,
- Ensure appropriate encryption of all data around binding.
- Explore the potential of variation in binding time as a covert channel.

# Choice of Technologies

- Ensure limitations of technologies are understood and the potential for future compromise is well identified.
- Ensure your chosen technologies support the tactics you want to deploy to protect the system.

# Summary

- We need to consider attacks on Confidentiality, Integrity and Availability.
- Attacks need to be monitored and detected, resisted where possible, otherwise we may need some other form of reaction, eventually we should be able to fully recover.
- Tactics are a high-level way of categorizing possible protection against attack.
- In this area there is need for considerable domain expertise, see for example the NIST Framework for infrastructure cybersecurity:
  - http://www.nist.gov/cyberframework/upload/cybersecurity-framework-021214.pdf