Demonstration of an Intrusion-Tolerant IDS

MAFTIA WP3

Dominique Alessandri

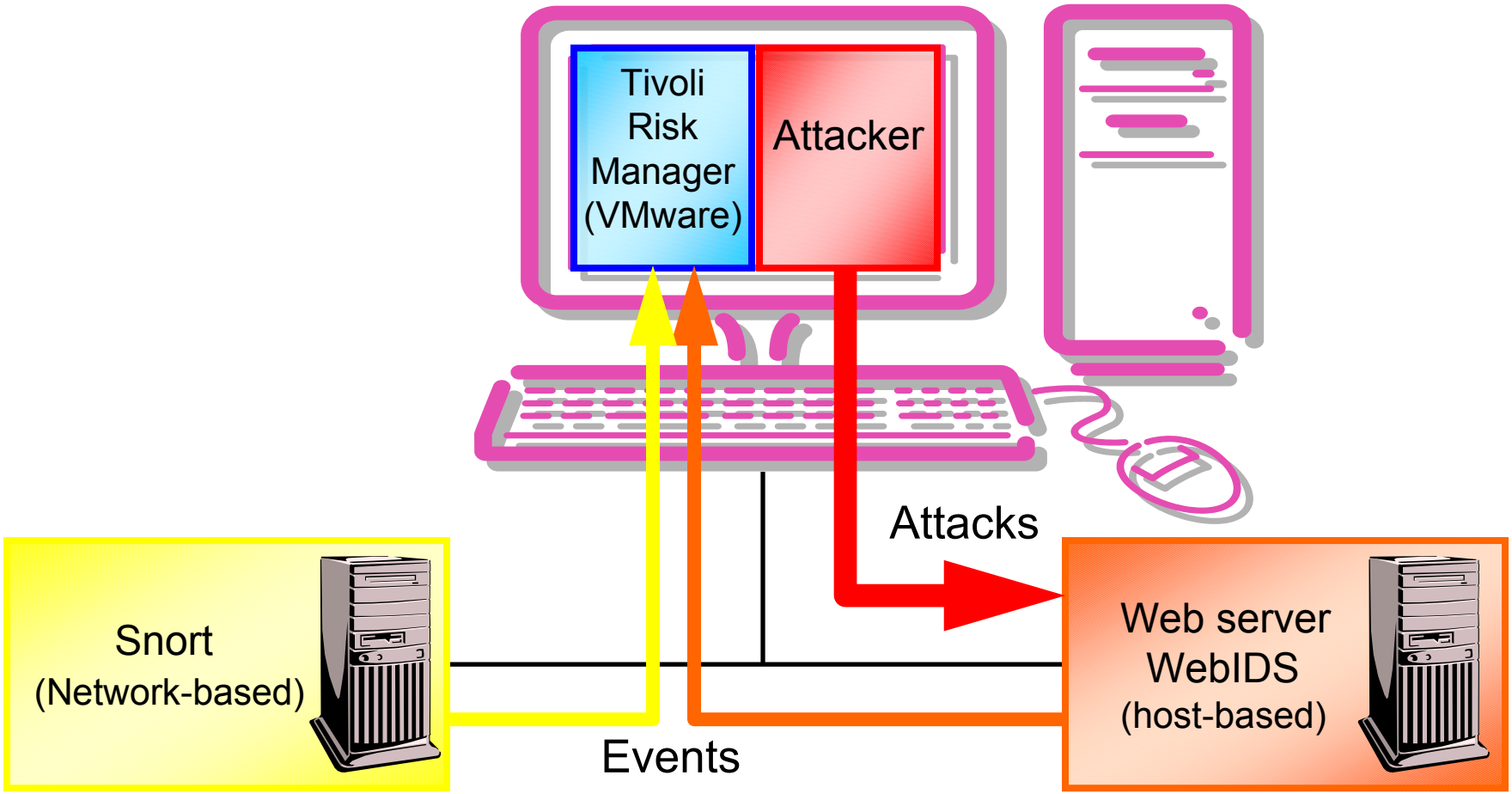
IBM Zurich Research Laboratory

Newcastle February 18 & 19, 2003

Overview

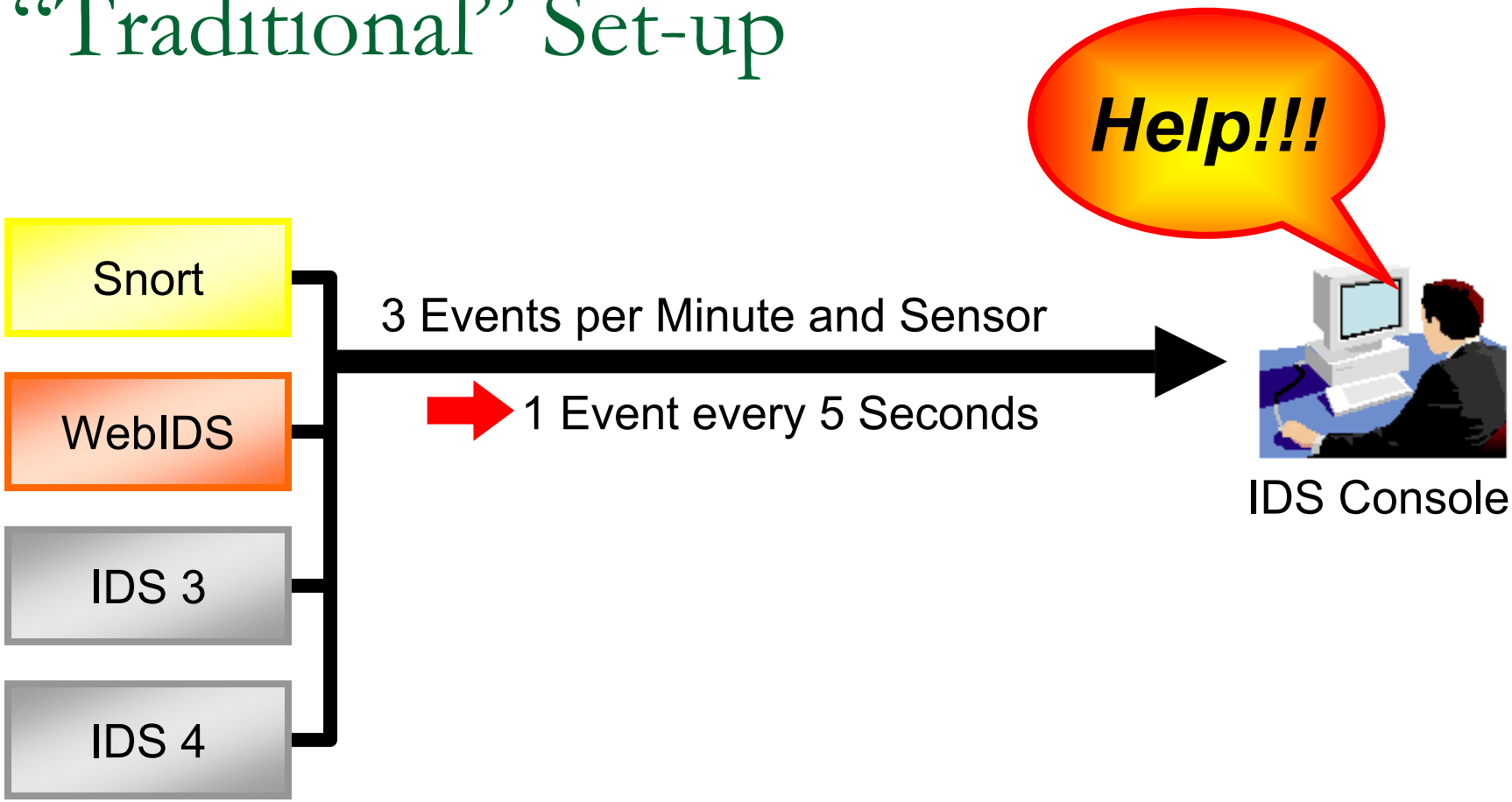
- Demonstration Set-up
- Demos
  - Filtering of false alarms
  - Reliable replication of event correlation
  - Self-diagnosing IDS

Demonstrator Components: Overview of Setup

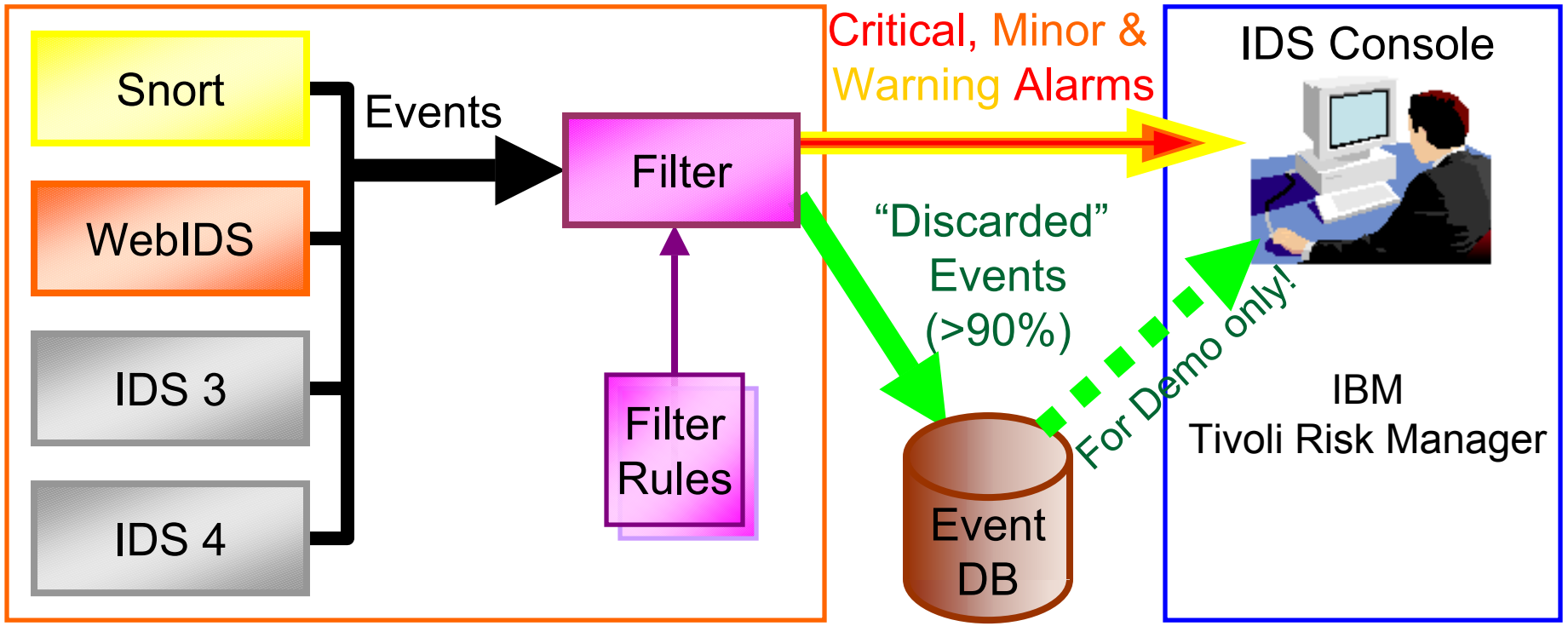


Demo 1:
Filtering of False Alarms

Discarding of False Alarms: "Traditional" Set-up

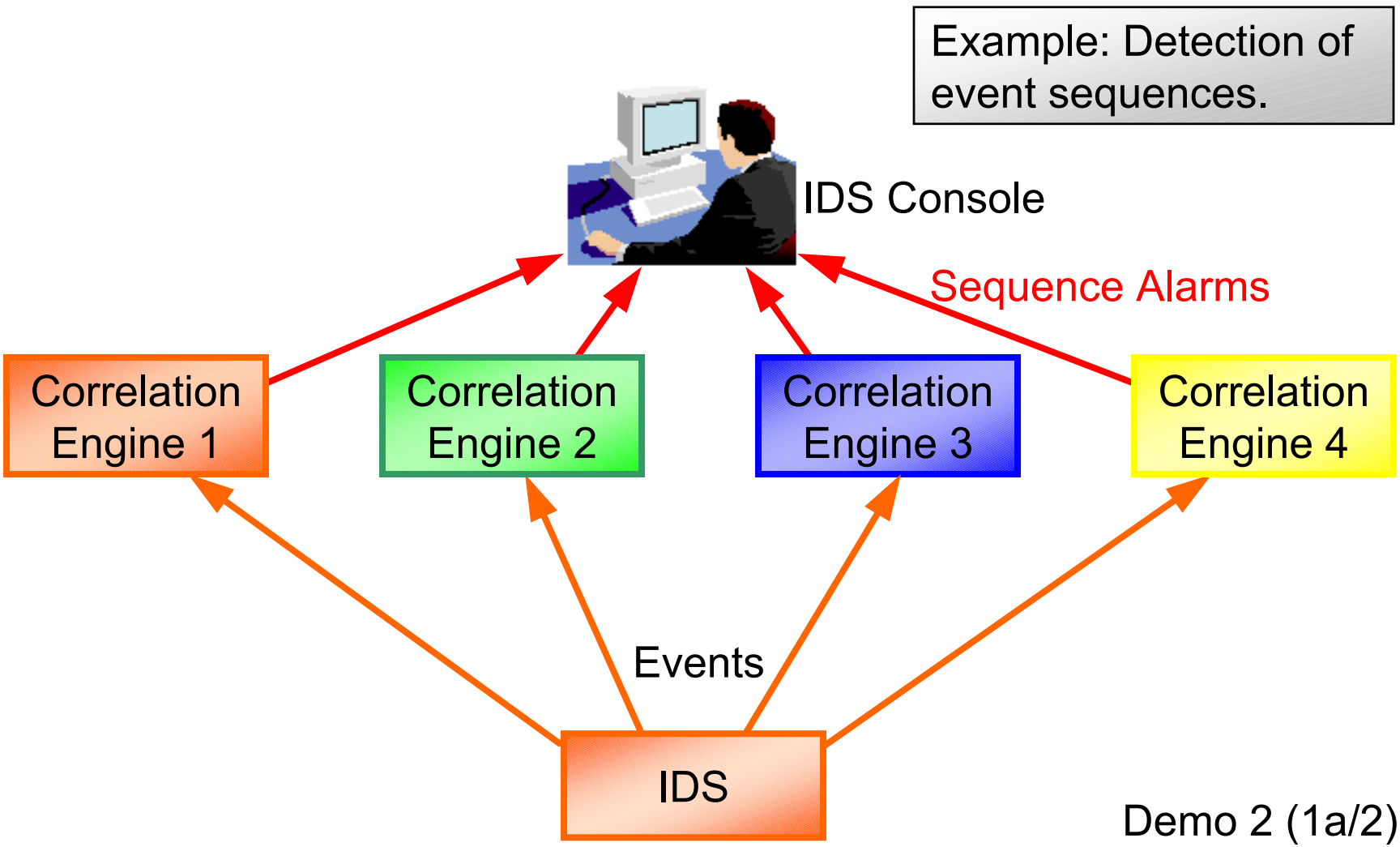


Demo: Discarding of False Alarms



Demo 2: Reliable Replication of Event Correlation

Replication of Correlation Engines



Demo 2 (1a/2)

Replication of Correlation Engines

Problem: Owing to operational and / or malicious causes alarms may arrive out of order or be lost.

Example: Detection of event sequences.

IDS Console

Sequence Alarms

Correlation Engine 1

Correlation Engine 2

Correlation Engine 3

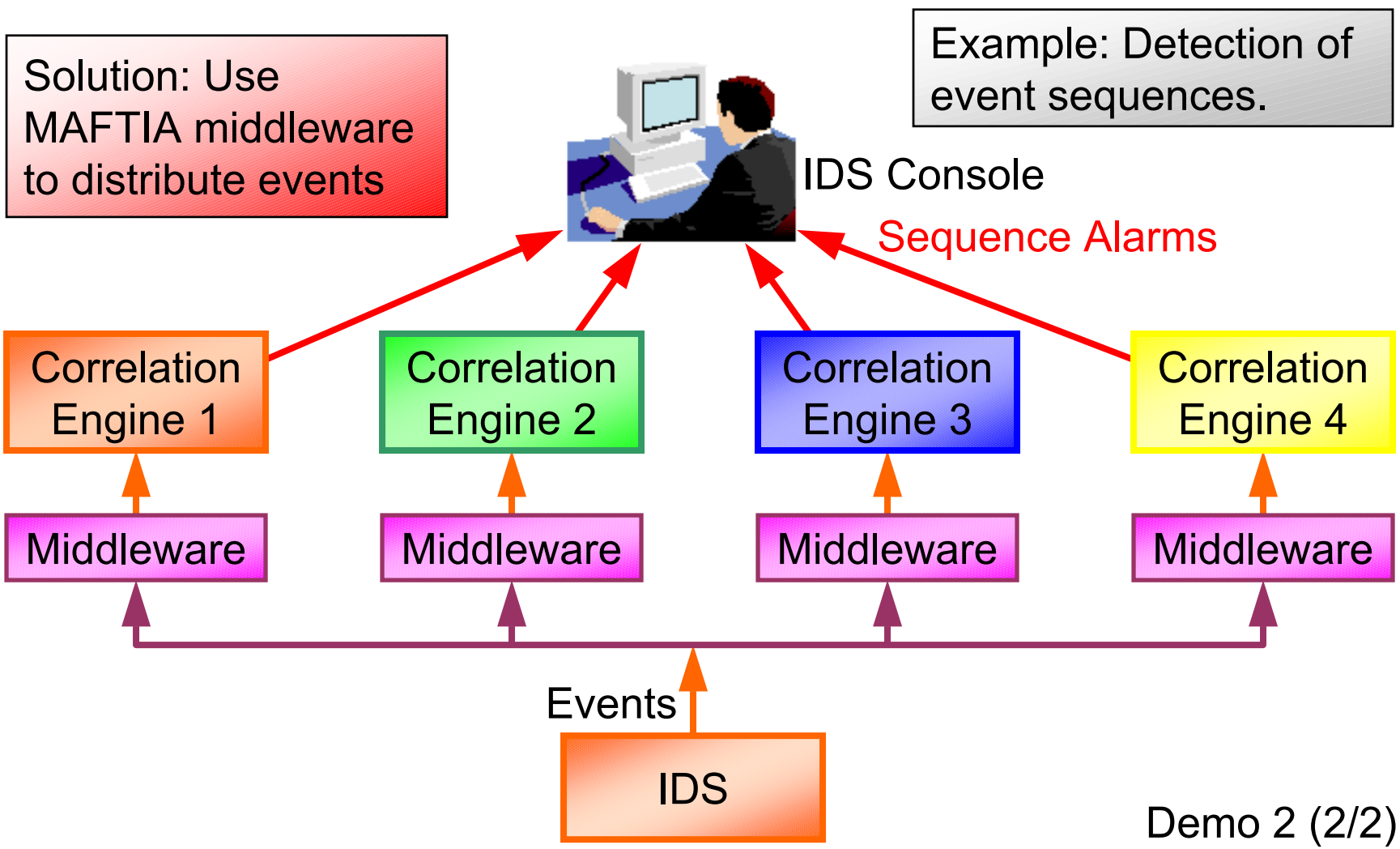
Correlation Engine 4

Demo: Random re-ordering & dropping of events

Events

IDS

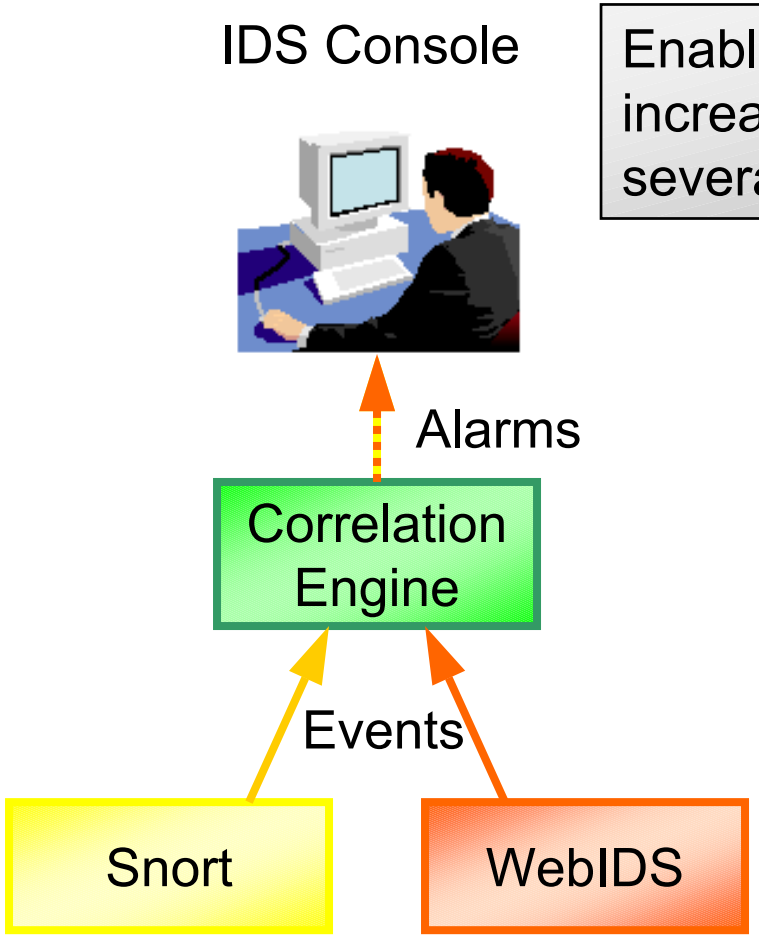
Replication of Correlation Engines



Demo 2 (2/2)

Demo 3:
Self-Diagnosing IDS

Self-Diagnosing IDS: Scenario 1



Enable self-diagnosis and increase reliability by using several IDS sensors.

Self-Diagnosing IDS: Scenario 1

Scenario 1:
Alarm Injection
(Confusion attempt)

Enable self-diagnosis and increase reliability by using several IDS sensors.

Requirement: Event correlation engine is aware of IDS capabilities and network topology

→ Correlation engine can detect inconsistencies

IDS Inconsistency Alarm

Alarms

Correlation Engine

Alarm Injection

Adversary

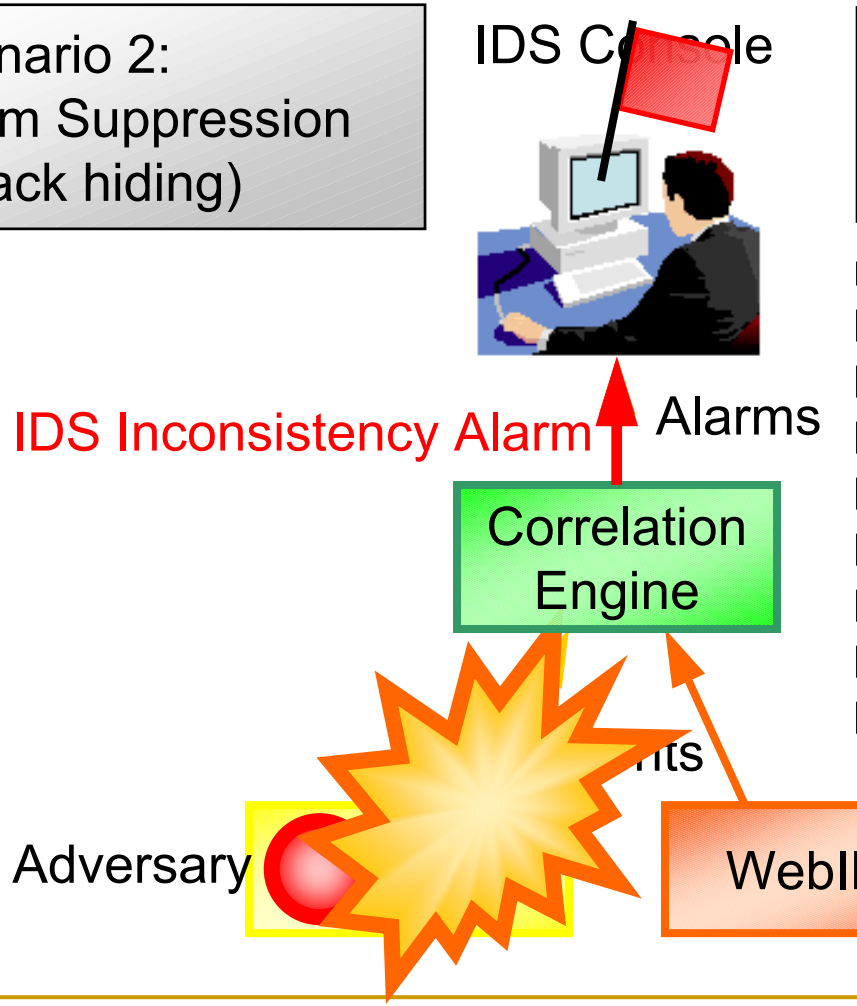
Events

Snort

WebIDS

Self-Diagnosing IDS: Scenario 2

Scenario 2:
Alarm Suppression
(Attack hiding)



Enable self-diagnosis and increase reliability by using several IDS sensors.

Requirement: Event correlation engine is aware of IDS capabilities and network topology

→ Correlation engine can detect inconsistencies