The Design of a Generic Intrusion-Tolerant Architecture for Web Servers

Ayda Saidane, Vincent Nicomette, and Yves Deswarte, Member, IEEE

IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL. 6, NO. 1, JANUARY-MARCH 2009

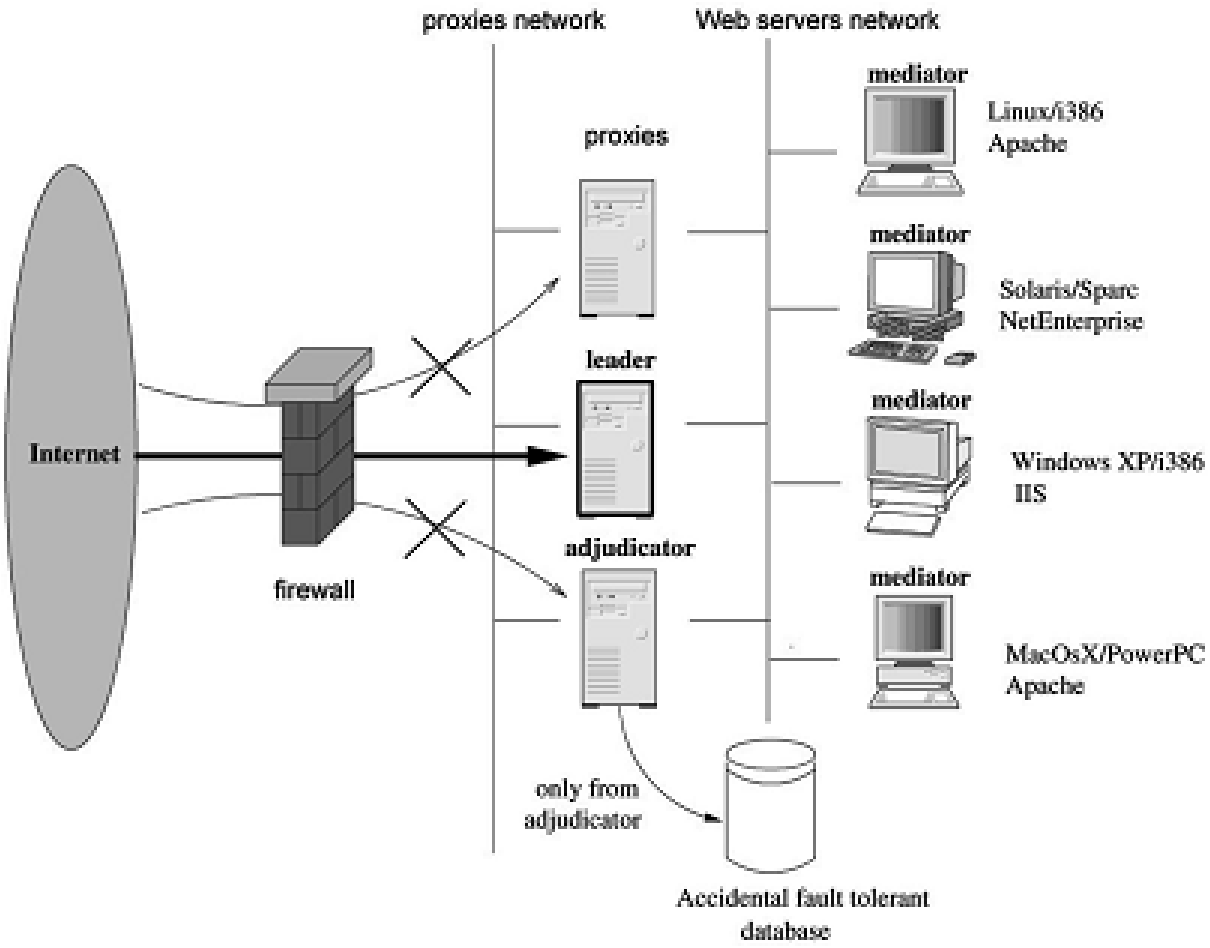
Outline

- INTRODUCTION
- AN INTRUSION-TOLERANT WEB SERVER
- DETECTION MECHANISMS
- ALERT MANAGER
- PERFORMANCE MEASUREMENTS
- CONCLUSION

INTRODUCTION

- Everybody agrees now that the Internet has become essential in everyday life.
- A growth of malicious activity in the Internet.
- More and more vulnerabilities are discovered, and nearly every day, new security advisories are published.

AN INTRUSION-TOLERANT WEB SERVER(1/4)



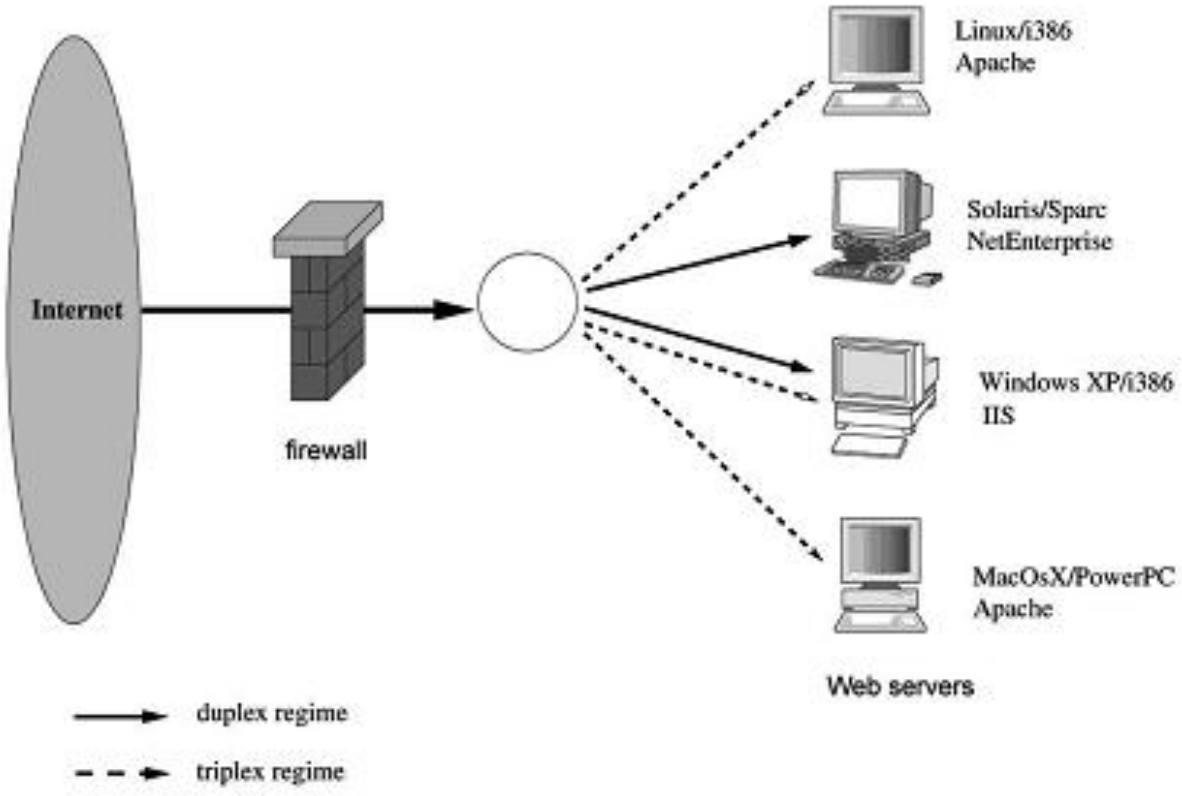
AN INTRUSION-TOLERANT WEB SERVER(2/4)

- The architecture is based on the principles of **redundancy** and **diversification**.
  - **Redundancy** is used to increase system availability.
  - **Diversification** is used to increase independence between redundant subsystems .

AN INTRUSION-TOLERANT WEB SERVER(3/4)

- **Adaptive Redundancy Level**
  - In order to minimize the performance degradation of the system.
  - The **regime** is the number of Web servers that process each client request.

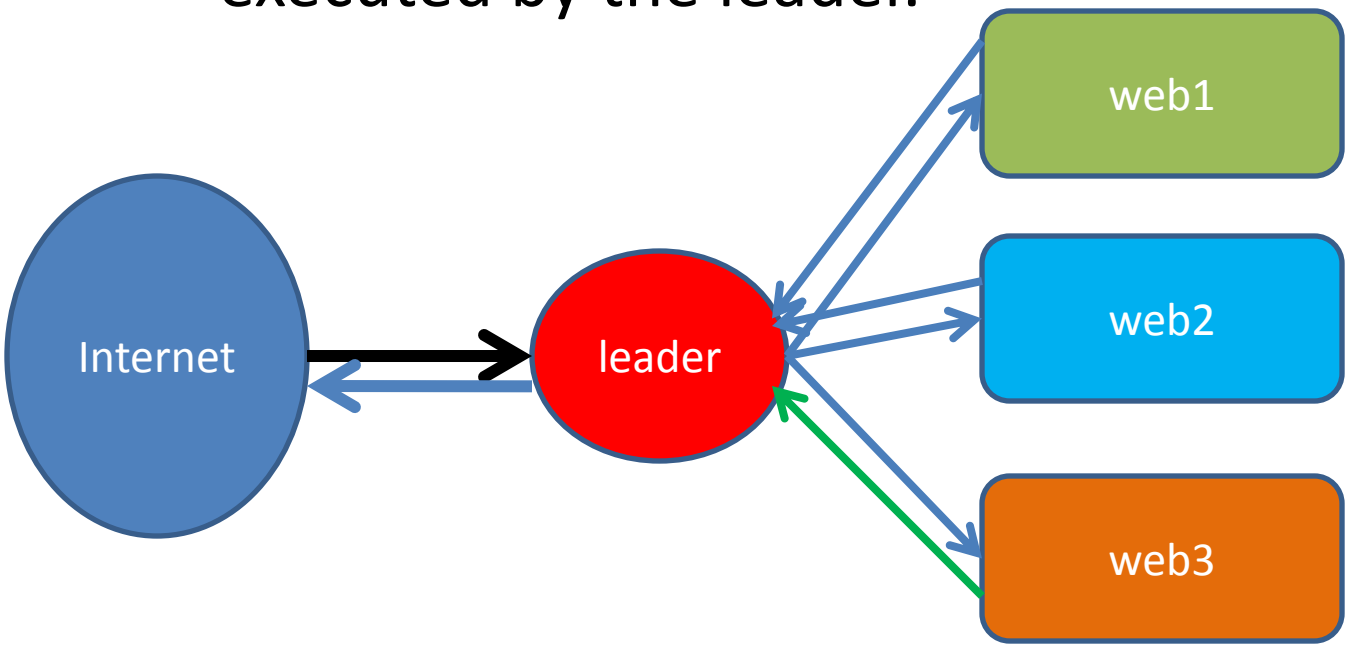
AN INTRUSION-TOLERANT WEB SERVER(4/ 4)



Duplex and triplex regimes.

Detection Mechanisms(1/8)

- **Agreement Protocol**
  - It is used to validate server responses when the system is running in a nonsimplex regime and is executed by the leader.



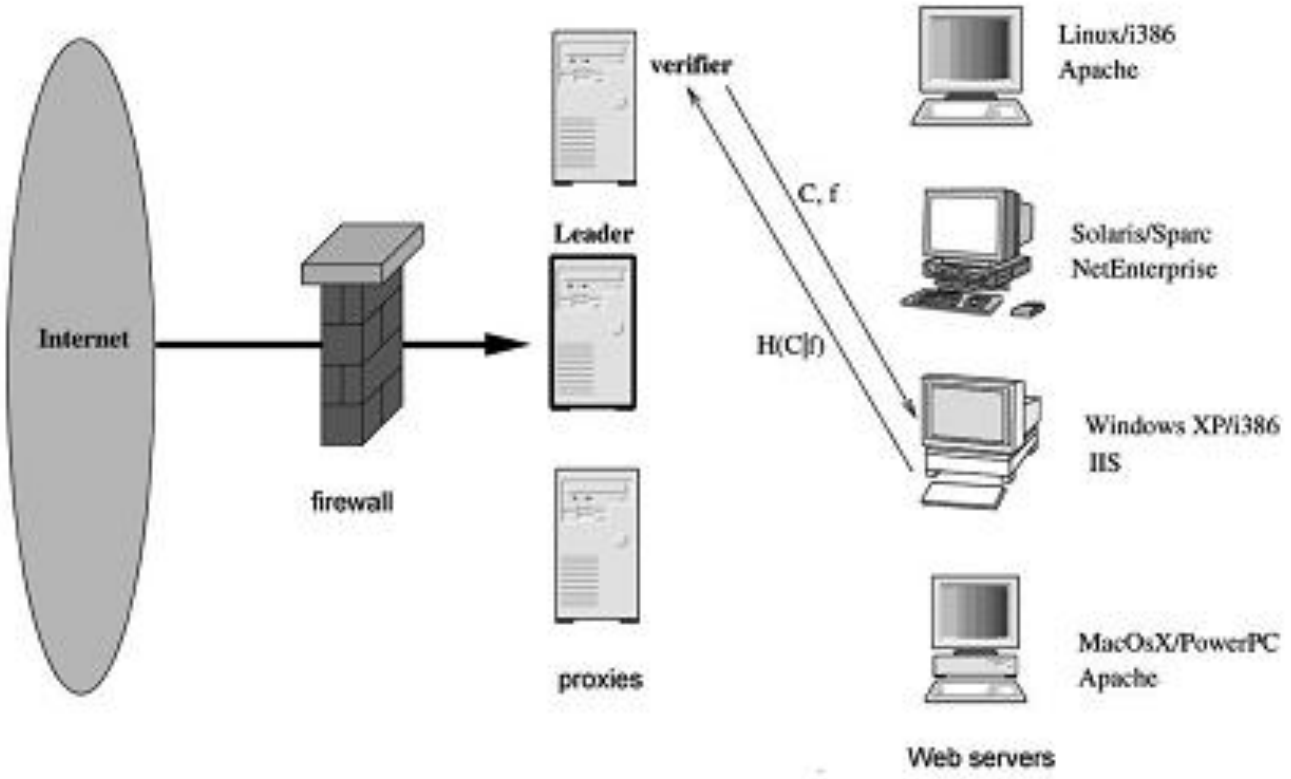
Detection Mechanisms(2/8)

- **Intrusion Detection**
  - **SNORT**
    - an open source software based on misuse detection.
  - **EMERALD**
    - developed at SRI International
    - which combines misuse detection and anomaly detection.

Detection Mechanisms(3/8)

- **Challenge/Response Protocol**
  - The IDSs are efficient in detecting known attacks but are less efficient in detecting new attacks with a slow propagation or low frequency.

Detection Mechanisms(4/8)



Challenge Response Protocol.

Detection Mechanisms(5/8)

- The number of challenges should be such that

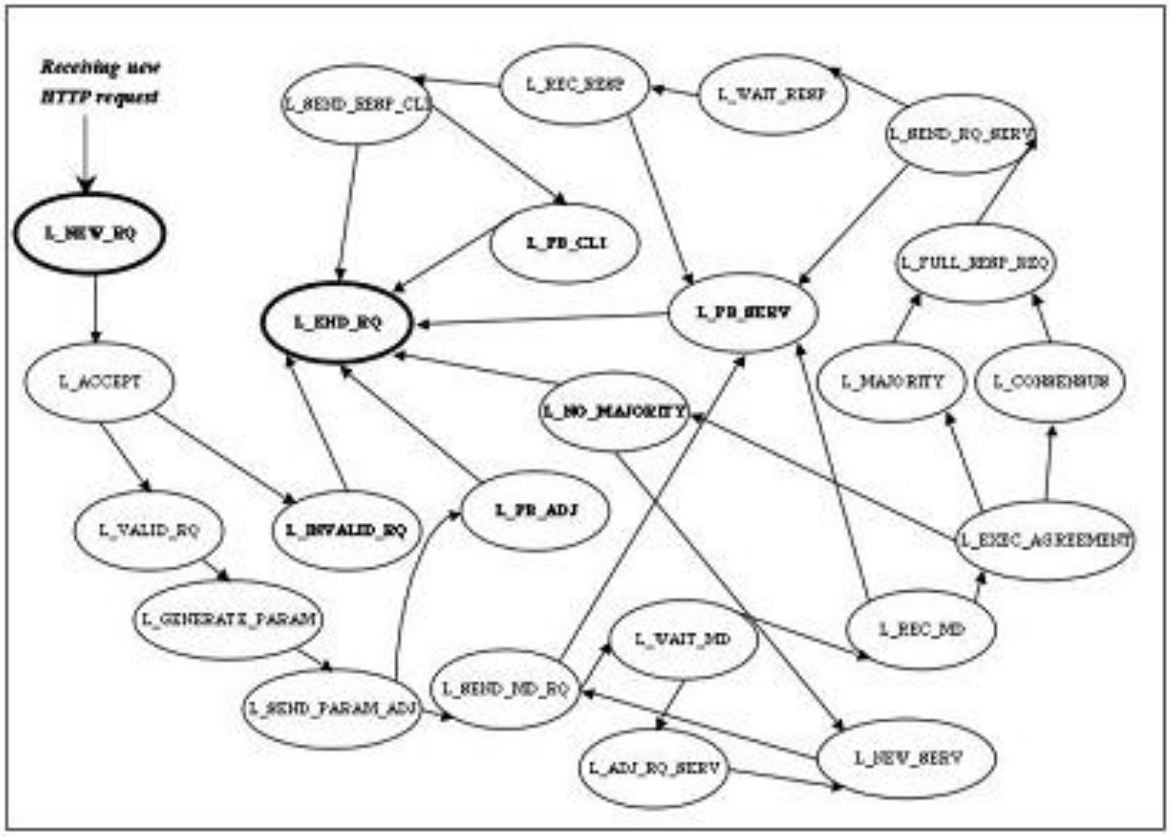
  M > fc / fr

  - **1. fc: the frequency of the CRP for the same file**
  - **2. fr: the reboot frequency.**

Detection Mechanisms(6/8)

- **Runtime Verification**
  - A runtime verifier checks the behavior of each proxy during its execution.
  - This technique detects with good credibility any injection of malicious code in the proxies.

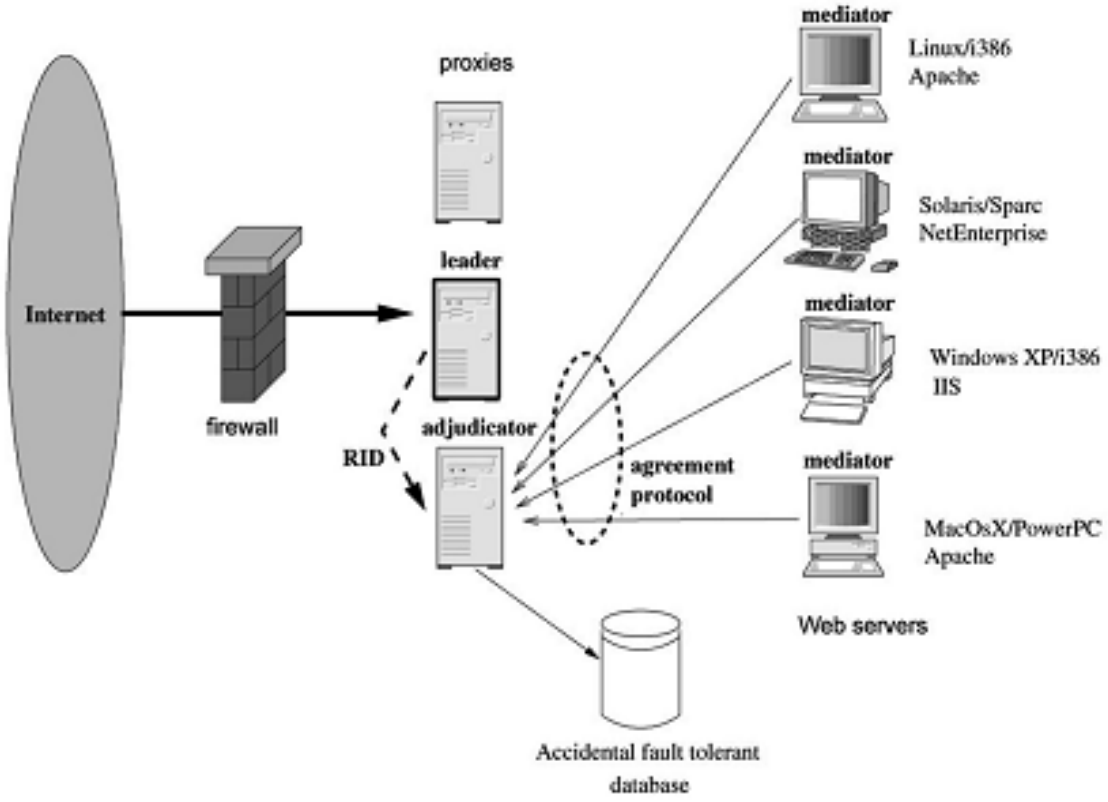
Detection Mechanisms(7/8)



State machine for the leader.

Detection Mechanisms(8/8)

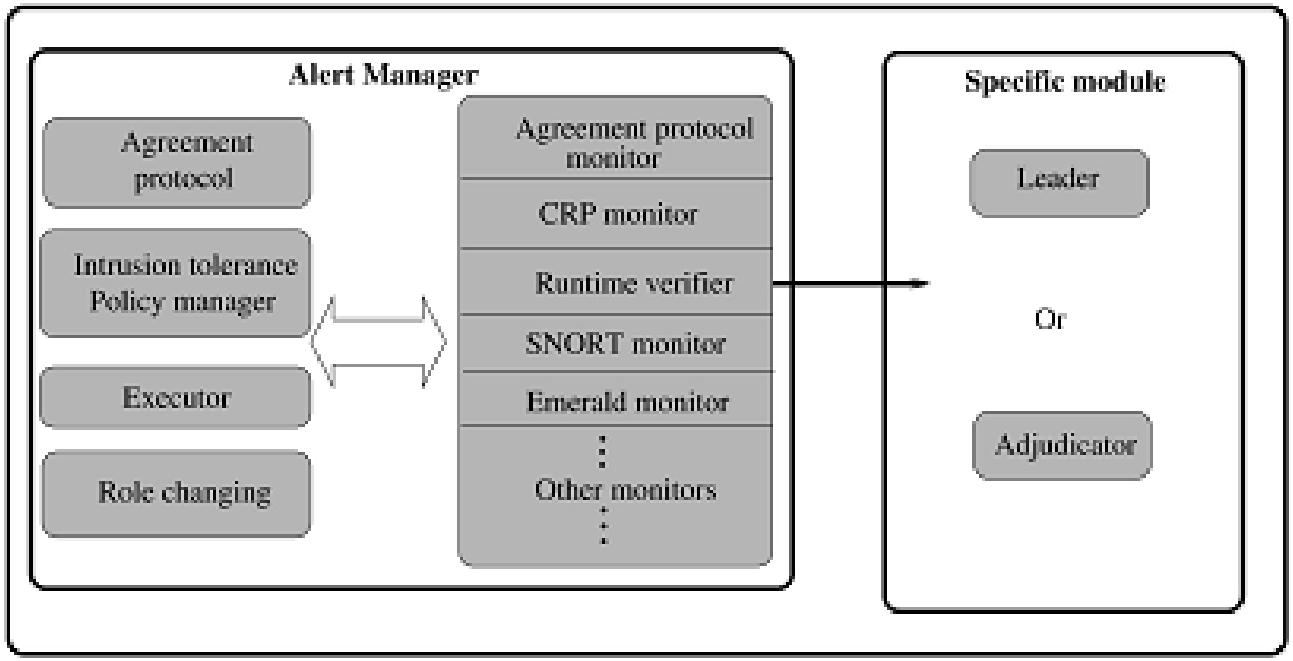
- **Database Access**



Connections between the leader, the adjudicator, and the mediator.

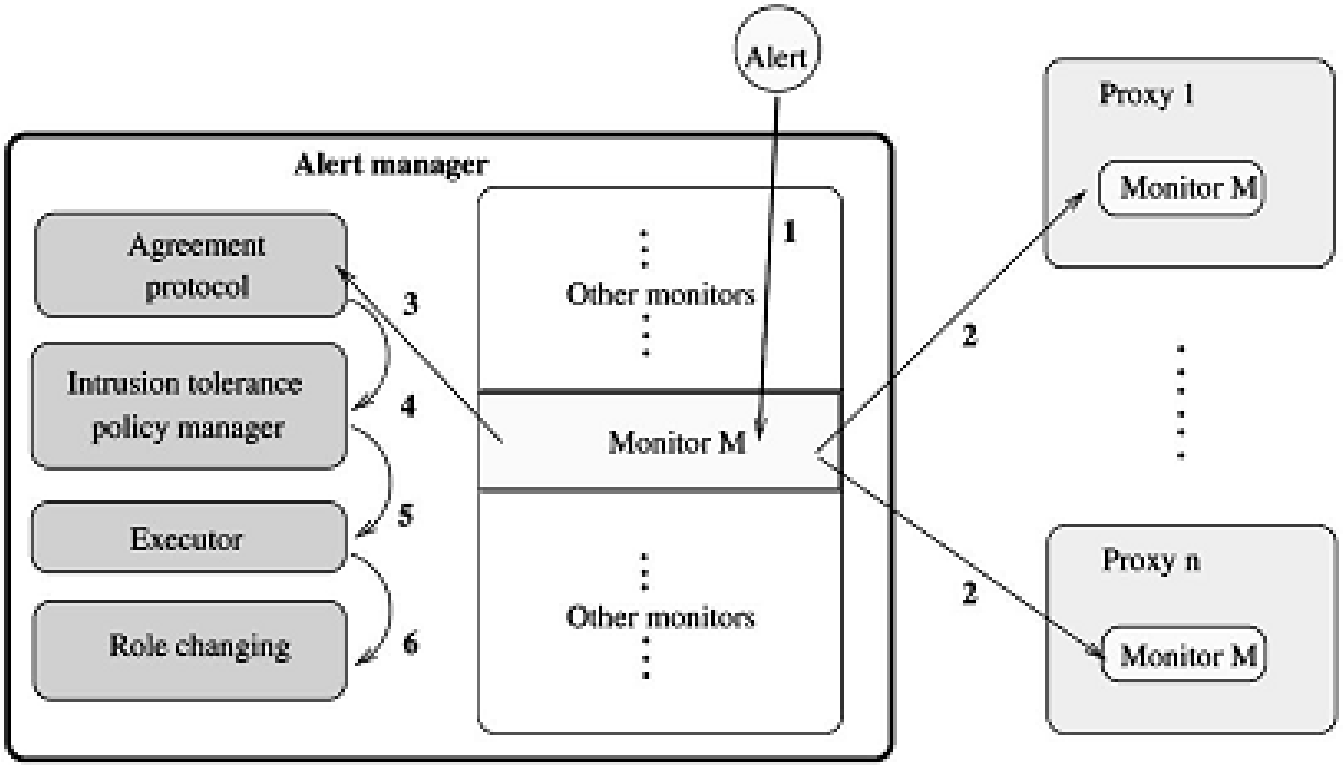
Alert Manager(1/2)

- illustrates the proxy architecture



Proxy architecture.

Alert Manager(2/2)



PERFORMANCE MEASUREMENTS(1/3)

- Performances According to the Regime

Global Processing Time According to the Regime and Size of the File

	Direct	Simplex	Duplex	Triplex
0 byte	0.0037 sec	0.0074 sec	0.0087 sec	0.0096 sec
44Kb	0.0115 sec	0.0145 sec	0.0167 sec	0.0170 sec
1Mb	0.14 sec	0.316 sec	0.321 sec	0.322 sec

PERFORMANCE MEASUREMENTS(2/3)

- Performance of Database Accesses

Comparison of Duration Using Our Library and MySQL Standard Library

	GPT_{DB}	APT (leader)	GPT_{HTTP}
using our database library	0.028 sec	0.038 sec	0.045 sec
using a standard MySQL library	-	0.015 sec	0.020 sec

PERFORMANCE MEASUREMENTS(3/3)

- Performance of Isolation and Reboot of a Corrupted Server

Details of Alert Processing

	APT	TI	GPT_A
Average	0.341 sec	0.344 sec	73.2 sec
Min	0.00616 sec	0.00876 sec	70.3 sec
Max	1.010 sec	1.015 sec	75.8 sec

CONCLUSION

- In this paper propose a generic intrusion tolerant architecture based on redundancy and diversification.

- The efficiency of intrusion tolerance is strongly dependent on the deployed detection mechanisms.