Computational Intelligence in Cyber Security

Dipankar Dasgupta, Ph.D

Professor, Computer Science

IEEE Senior Member

Intelligent Security Systems Research Lab
Dunn Hall, Rm 120
The University of Memphis
Memphis, TN 38152

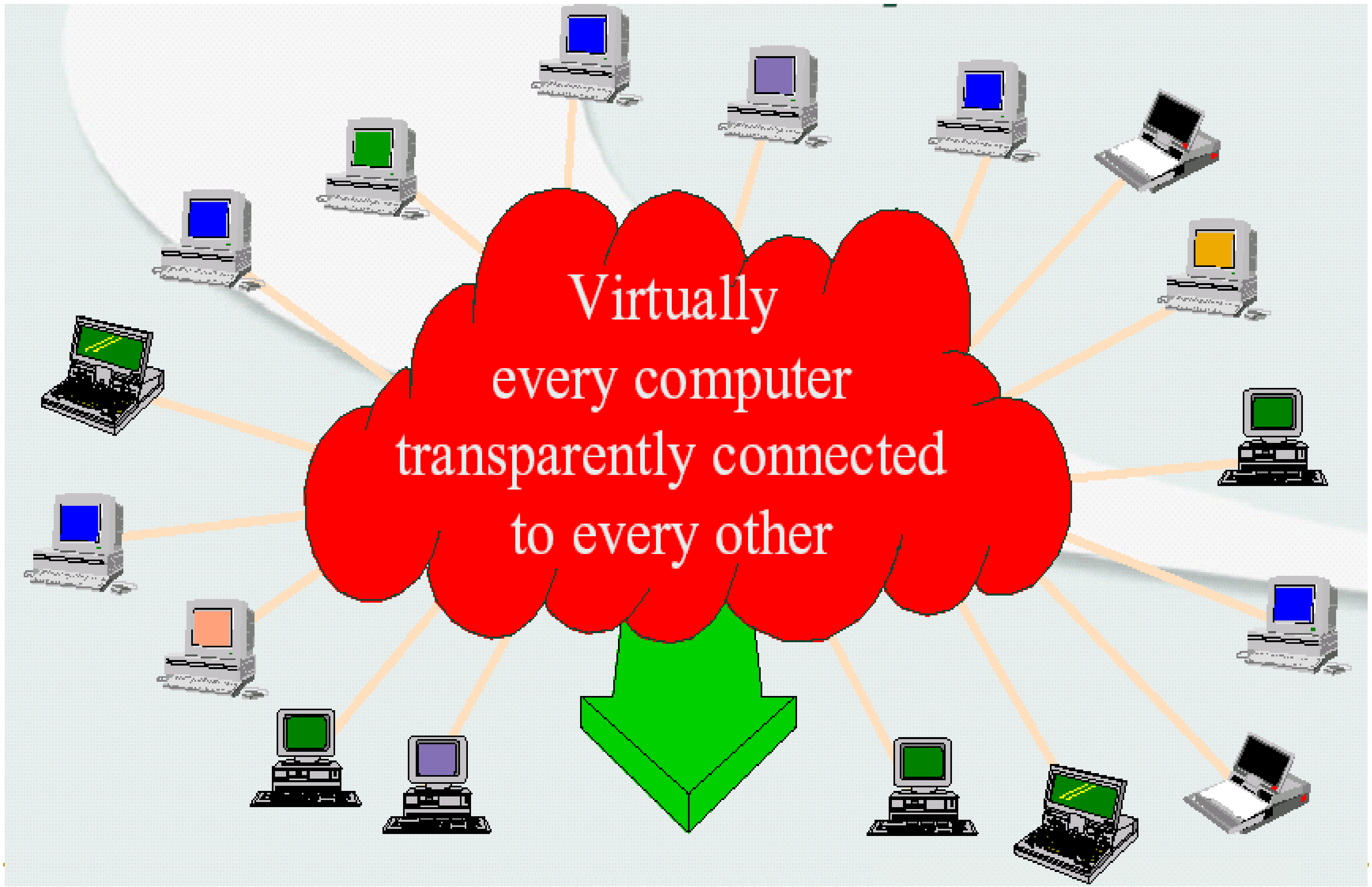
Director, Center for Information Assurance
FedEx Institute of Technology, 324
The University of Memphis
Memphis, TN 38152

THE UNIVERSITY OF MEMPHIS

Topics to be covered

- Cyber Space - Basics
- Cyber Security issues
- Cyber Defense Technologies
- New Security Challenges & Computational Intelligence solutions
- Intrusion Detection Approaches
  - Neural Networks
  - Fuzzy Logic
  - Evolutionary Algorithms
  - Fuzzy Clustering
  - Artificial Immune Systems
  - Cellular Automata
- References

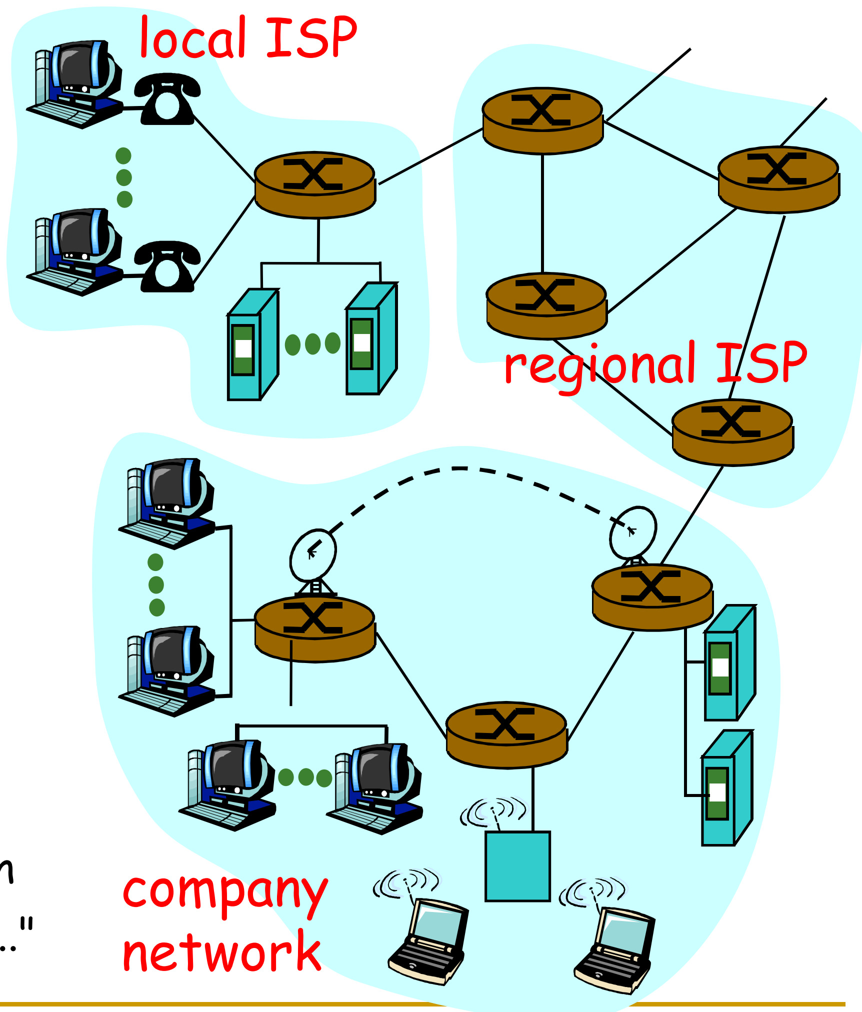
Cyber world: *Global view*



Cyber Space: *Interconnectivity*

- millions of connected computing devices: *hosts, end-systems*
  - pc's workstations, servers
  - PDA's phones, toasters

  running *network apps*
- *communication links*
  - fiber, copper, radio, satellite
- *routers:* forward packets (chunks) of data thru network



❒ cyberspace: "a consensual hallucination experienced daily by billions of Internet users.."

Cyber Infrastructure

- Our Society is increasingly dependent on Internet and same is our mission-critical infrastructure:
  - **Telecommunications**
  - **Power**
  - **Finance & banking**
  - **Transportation**
  - **Commercial & other industrial activities**
  - **Military and Government operations**
- However, Internet's underlying structure, protocols, & governance are still primarily open

INTERNET: *Scope, Benefits and Dangers*

- Brings people together
- Makes world seem smaller
- opens up new opportunities
- increases exchange of ideas and information
- Greater danger of harm on greater scale
    - Technology has made fraud easier for hackers and criminals
    - Fraud kept pace with the rising popularity of online business --Thefts of credit card numbers
    - Online anonymity makes fraudulent user more bold
    - Fraud protection increase the cost of doing e-business

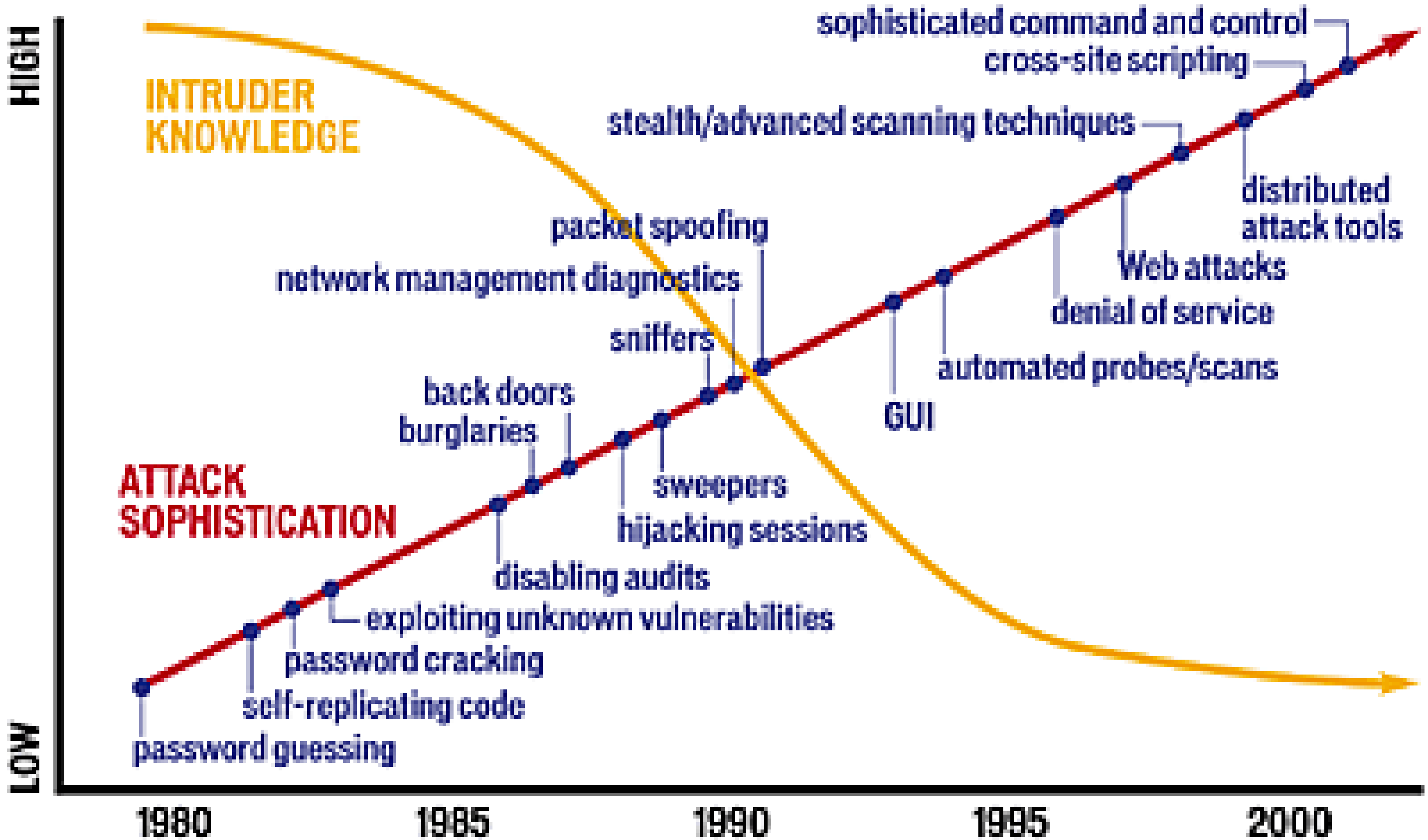
Proliferation of Wireless

- Ease and speed of deployment: Basic wireless network is easy to set up
- Inexpensive: Does not require expensive cabling infrastructure
- Scalable: Can be used to either extend an existing wire network, or build a new network
- Flexibility: No cabling and re-cabling
- Mobility and ease of access

Major Security Challenges:

- **Isolation and physical protection no longer adequate/appropriate/feasible**
- **Geographic spread**
  - remote access
  - sharing data & files across distance
- **User-user threat model no longer adequate**
- **Vulnerabilities**
  - accidental disclosure
  - deliberate penetration
  - active infiltration
  - passive subversion

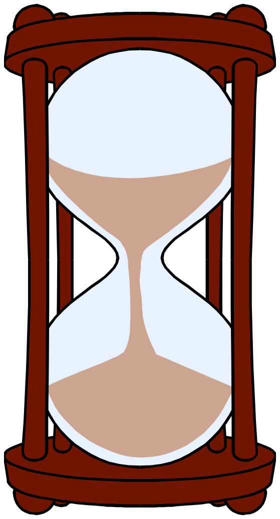
As the sophistication of Internet attacks increases, the technical knowledge of attackers on average is declining. Sophisticated attackers are building tools that novices can invoke with the click of a mouse.



Source: CERT Coordination Center, CMU, Pittsburgh

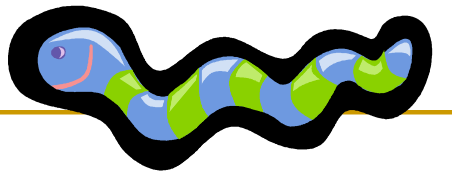
Cyber Threats

- Out-of-the-box Linux PC hooked to Internet, not announced:
- [30 seconds] First service probes/scans detected
- [1 hour] First compromise attempts detected
- [8 hours] PC fully compromised:
  - Administrative access obtained
  - Event logging selectively disabled
  - System software modified to suit intruder
  - Attack software installed
  - PC actively probing for new hosts to intrude



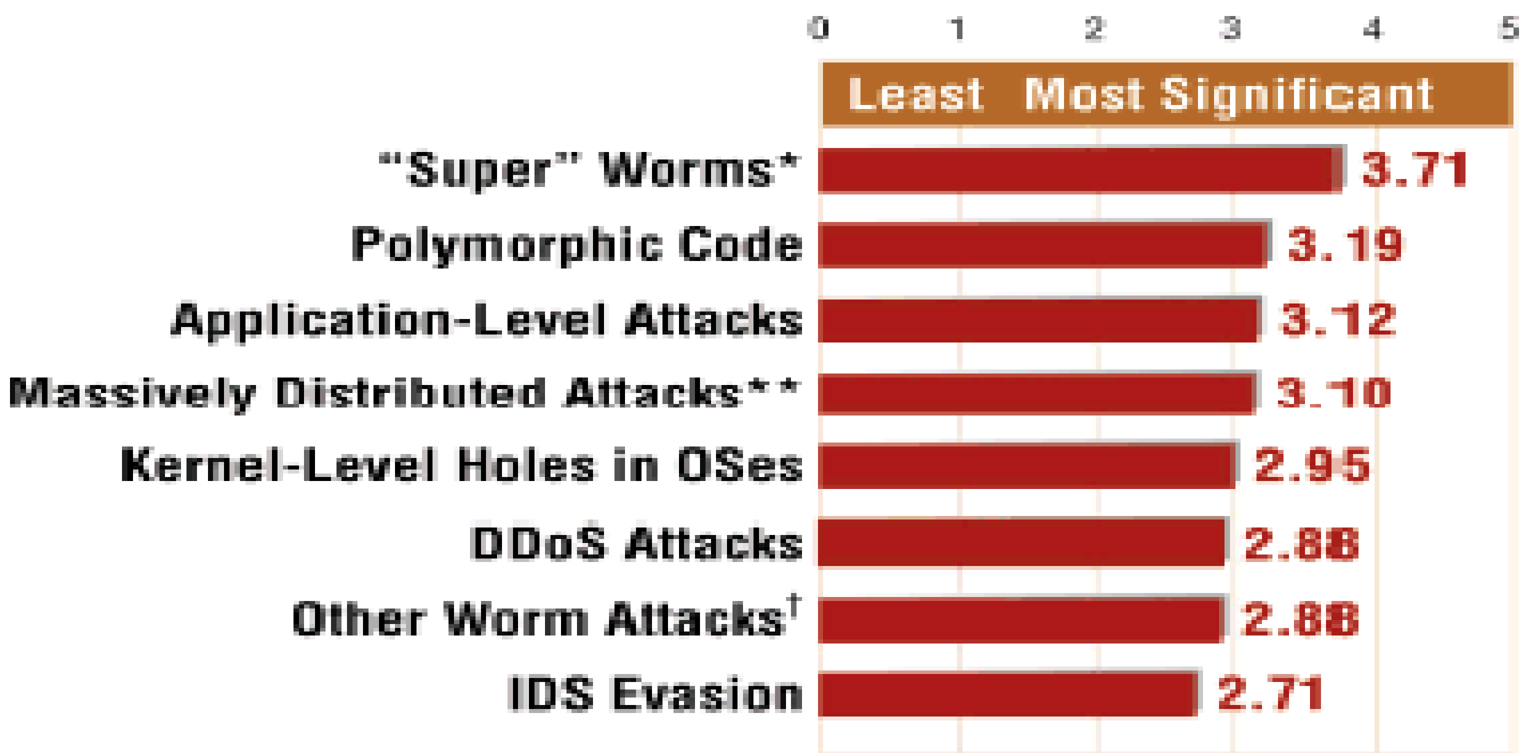
Cyber Threats

- *Identity Theft* which is reaching epidemic proportions;
- ▪*Cyber-hacking* which is continually front-page news; and
- ▪*Malware, worm,Virus threats, Phishing, Botnet, Spam* appear almost daily.
- Viruses are now
  - Intelligent, and can learn new exploits on the fly
  - Polymorphic, to avoid signature detection
  - Programmable, to learn vulnerabilities and be remotely controllable



SECURITY THREATS, 2003-2008

What will be the biggest security threats over the next five years?



*Fast-spreading, multiplatform, multi-exploit, zero-day, metamorphic worms.

**Beyond traditional DDoS, including distributed password cracking, distributed port and vulnerability scanning, etc.

†Worms spreading beyond Internet—e.g., to telephony, power grid.

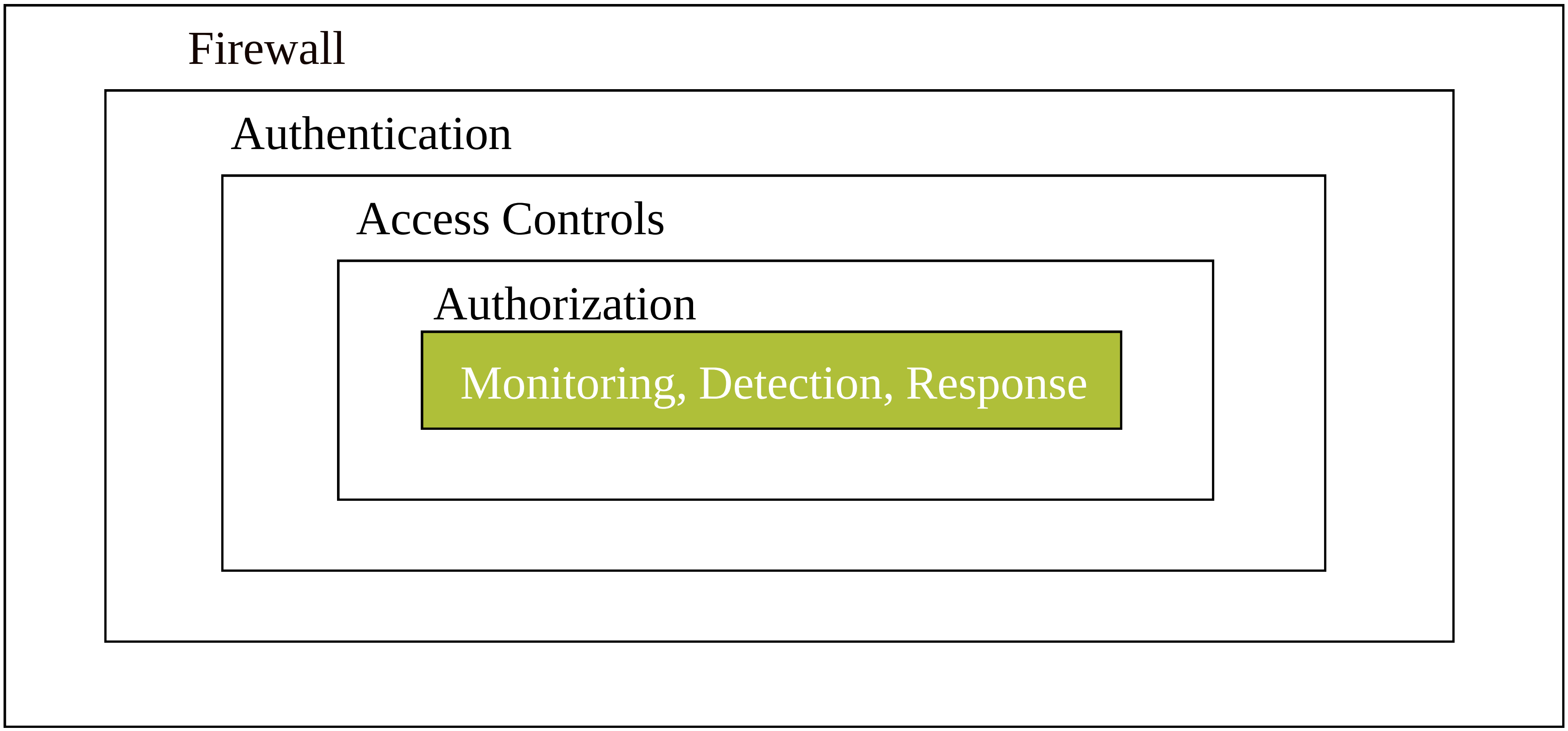
Source: *Information Security* survey of 220 readers.

Security Goals

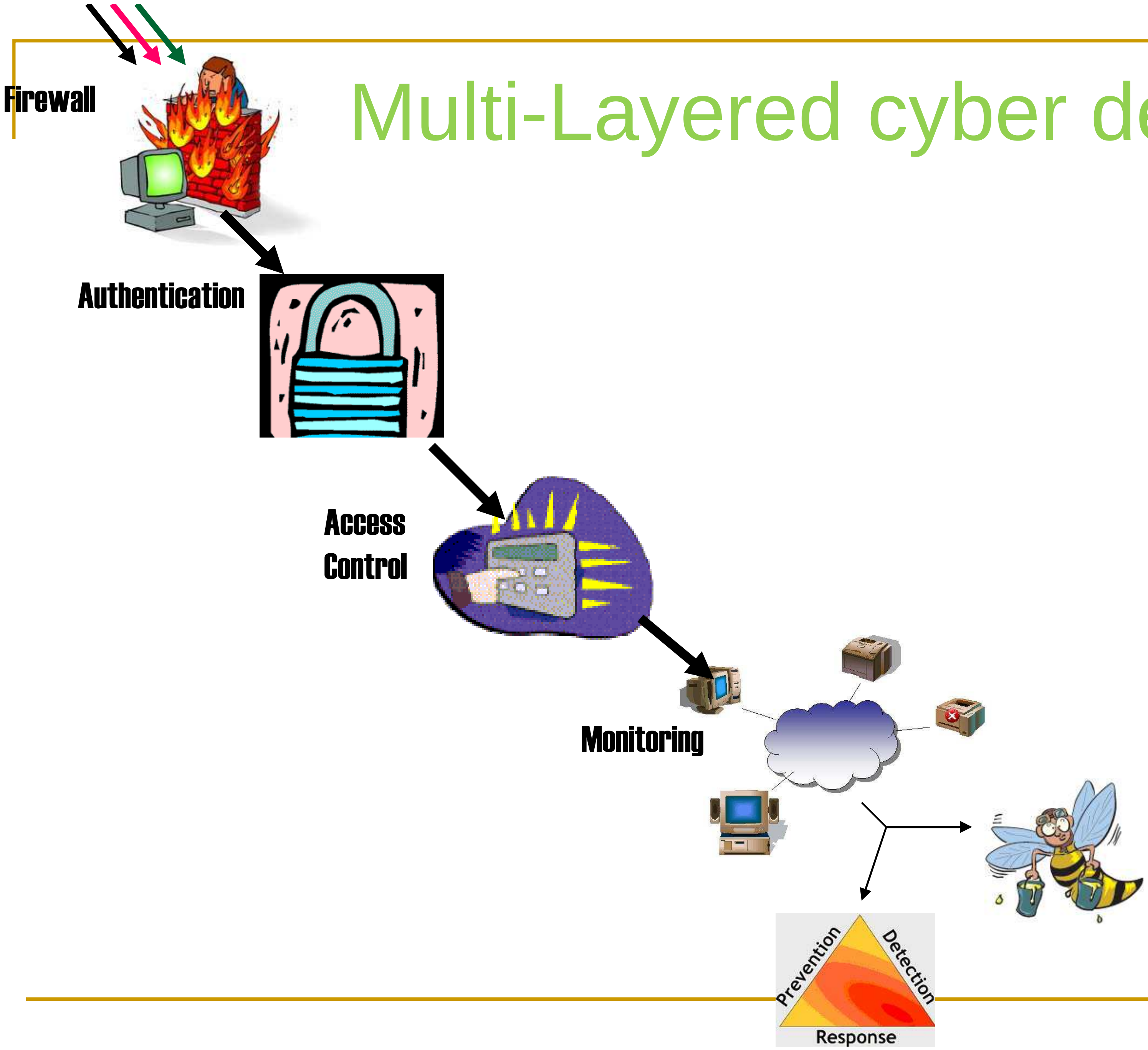
Three key qualities that information security seeks to ensure ("CIA")

- Confidentiality
  - private data should be known only to the owner of the data, or to a chosen few with whom the owner shares the data
- Integrity
  - the system and its data must be complete, whole and in readable condition, precise, accurate
- Availability
  - the system must be available for use when the users need it. Similarly, critical data must be available at all times

Multi-Layered Security (Javitz,1992)



Multi-Layered cyber defense



Intrusion Detection (ID)

It is the process of monitoring the events occurring in a computer system or network and analyzing them for signs of intrusions, defined as attempts to bypass the security mechanisms of a computer or network ("compromise the confidentiality, integrity, availability of information resources")

- Misuse Signature Vs. Anomaly (or behavior profile) Based Detection
- Network-Based Vs. Host-Based Detection
- Real-time IDS Vs. Off-line IDS
- Hybrid Approaches

Signature (or Misuse) Based ID

Scan packets, logs, commands for known malicious patterns (*pattern matching*)

- ***Advantages:***
  - They have a potential for very low alarm rates
  - Easier for the security officer to take preventive or corrective action
- ***Disadvantages:***
  - Difficulty in gathering the required information on the known attacks
  - Detection of insider attacks involving an abuse of privileges is difficult because in most cases no vulnerability is actually exploited by the attacker
  - Unable to detect new type of attacks

Anomaly (or Behavior) Based ID

Detect intrusions by developing statistical model of normal usage

- ***Advantages:***
  - They can detect attempts to exploit new and unforeseen vulnerabilities
  - Detect “abuse of privilege” types of attacks
- ***Disadvantages:***
  - High false alarm rates
  - Need periodic updating to accommodate legitimate changes in the system

Network-Based vs. Host-Based Intrusion Detection

Network-based

- Scans network packet logs for signatures of intrusive activities.
- Increasing bandwidth is a challenge.
- End-to-end encryption could obsolete this approach.

Host-based

- Scans machine audit logs for signatures of intrusive activities.
- Traditionally monitors users' behavior.
- Many sensors/hosts require enterprise management.

Real-time IDS Vs. Off-line IDS

Real-time IDS

- Analyzes the data while the sessions are in progress (e.g. network sessions for network intrusion detection, login sessions for host based intrusion detection)
- Raises an alarm immediately when the attack is detected

Off-line IDS

- Analyzes the data when the information about the sessions are already collected –post-analysis
- Useful for understanding the attackers' behavior

Responses on Intrusion Detection:

- ***Passive Alerting***
  - An alarm is generated when an attack is detected
  - Send email, pop-up messages
  - No action is taken in response to the attack

    Ex: send alert to log-file, create alert report

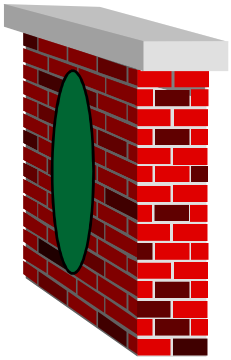
- ***Active Response***
  - Take countermeasures to revert to the former state in the event of abnormality
  - Trace route
  - Terminate the connection carrying an attack

Limitations of Existing IDSs

- Attacks usually occur both internally and externally
- External attacks rarely follow an expected patterns
- Attackers often work in concert
- Changes to existing network configuration can adversely effect IDS performance
- Attacks may occur over an extended period of time
- Once an intrusion is detected, systems need to identify, alert, isolate, and respond according to local security policies.

Limitations of other Defenses

Cyber attacks:



- Go through firewalls unimpeded
- Go unnoticed by intrusion detection systems
- Propagate too fast for anti-virus vendors to disseminate signatures in time
- Have complete access to network and file systems
- Execute with owner privileges
- Can send sensitive information out over networks
- Can spy on our computer and Web usage patterns

New Cyber attacks --New Thinking

Consider:

- Intrusion detection techniques are designed to handle Internet and network-based attacks
- Anti-virus software is designed to address malicious code attacks

- But, neither handle coordinated attacks effectively
- We need to either learn from the strengths of these approaches, or to develop a new approach entirely

Addressing cyber security challenges:

- For detecting a wide variety of
    - Active and passive attacks
    - External attacks and internal misuses
    - Known and unknown attacks
    - Viruses and spam
- We need flexible, adaptable and robust cyber defense system which can make intelligence decisions (in near real-time) while performing
    - Proactive and Reactive defense
    - Active and passive surveillance
    - Real-time and Off-line Analysis
    - Survivable systems

Computational Intelligence (CI)

The CI field of interest includes (but not limited to) the theory, design, application, and development of **biologically** and **linguistically** motivated computational paradigms emphasizing neural networks, connectionist systems, evolutionary computation, fuzzy systems, and hybrid intelligent systems in which these paradigms are contained.

CI Techniques in cyber security

- Main techniques used
  - Neural Networks
  - Fuzzy Logic
  - Evolutionary Algorithms
  - Gravitational Clustering
  - Cellular Automata
  - Artificial Immune Systems
  - Intelligent/Autonomous/Mobile Agents
- Issues with use of CI
  - Scalability
  - Sensitivity to parameters
  - Robustness

A Neural Network Approach in the Detection of Misuse : Initial Results (James Cannady, Presented at RAID '98)

Observations:

Ability to identify collaborative/temporally dispersed attacks

Review large data sets for patterns of activity

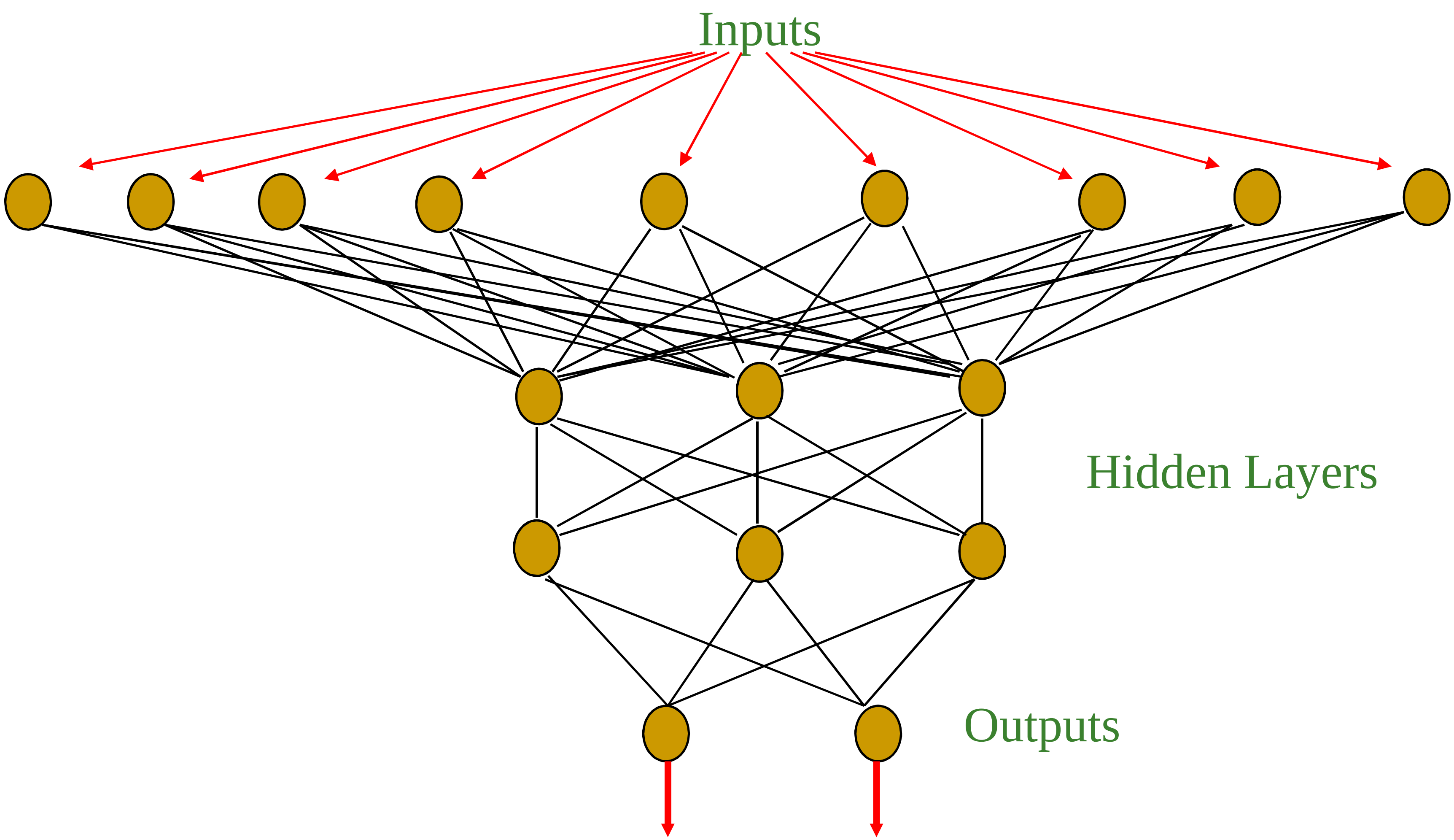
Analytical Speed

Properly designed neural networks are inherently fast

Sentinel: A Neural Network Approach

- Unlike previous NN-based approaches, Sentinel focused on the detection of instances of misuse
- RealSecure™ was used with Internet Scanner™ to generate and collect "attack" events
- NN architectures implemented with NeuralWorks Professional II/Plus™ from NeuralWare
- Two prototypes (experiments) were designed to test approach

Multi-layered Neural Networks

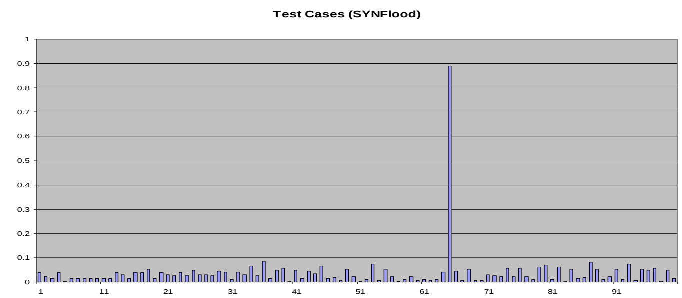


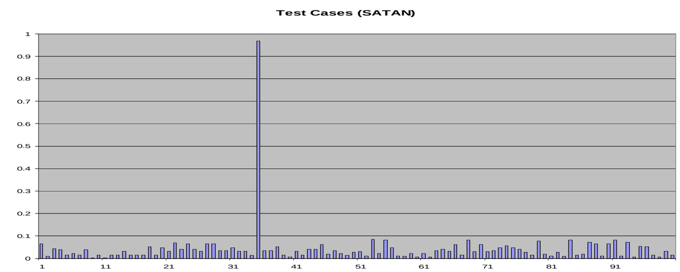
Sentinel: Prototype #1

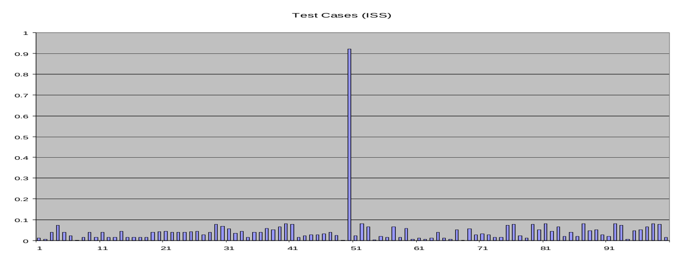
- NN nodes applied sigmoid transfer function (**1/(1 + exp (-x))**) to connection weights
- 10,000 events collected from network (~3,000 “attacks”)
- Preprocessing of data
  - Components selected from packets
    - protocol ID, source port, destination port, source address, destination address, ICMP type, ICMP code, raw data length, raw data
  - Conversion of some components (ICMP type, ICMP code and raw data) into normalized format
  - Addition of output fields (0/1)
  - Storage in database
- 10,000 iterations through NN/~9,000 training samples and 1,000 test examples

Sentinel: Prototype #1 Results

- Training/Test Results
  - Training data root mean square error = 0.058298
  - Test data root mean square error = 0.069929
  - Training data correlation = 0.982333
  - Test data correlation = 0.975569
- NN tested with limited streams containing ISS scan, SYN Flood, and Satan scan events





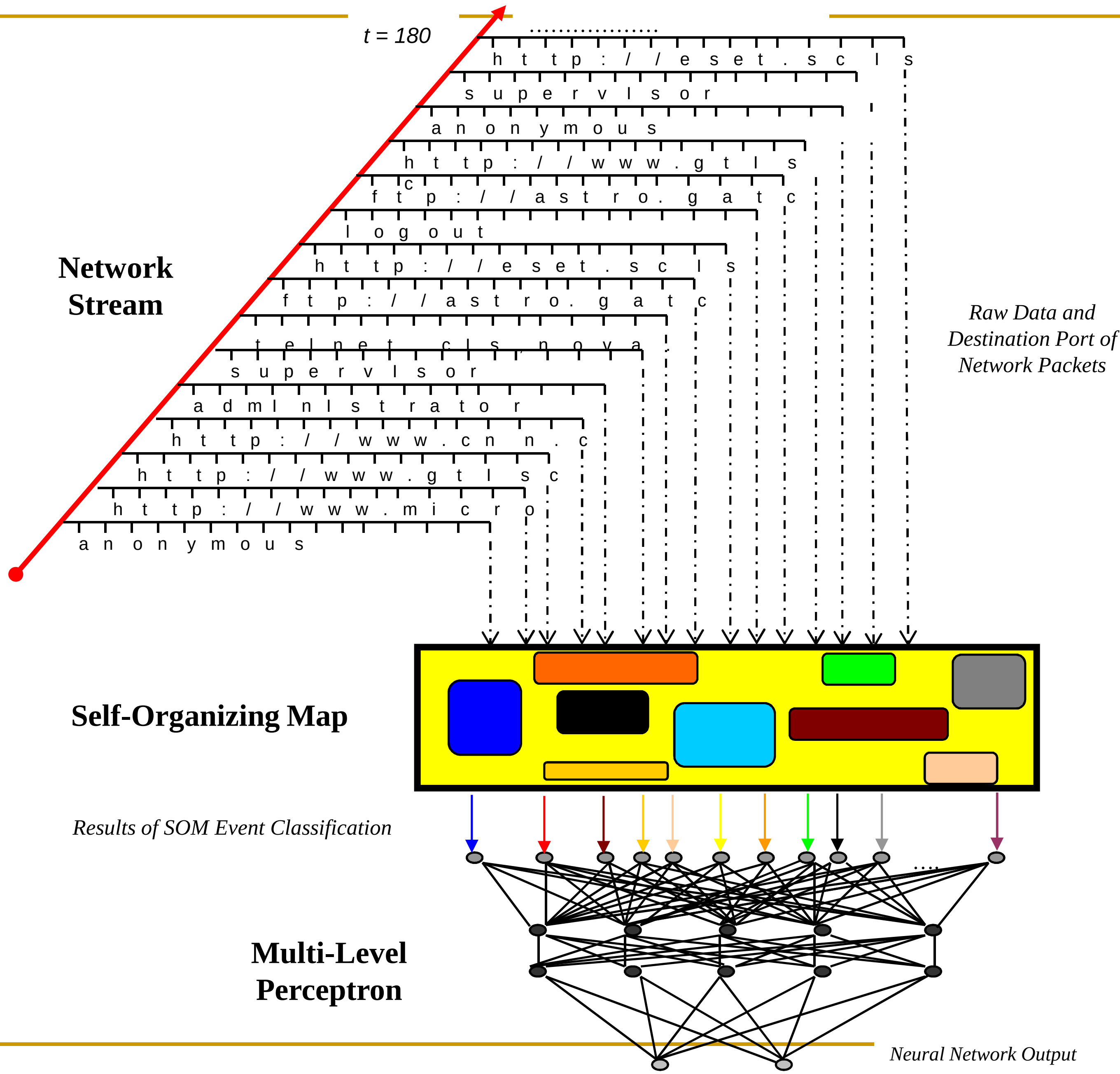


Sentinel: Prototype #2

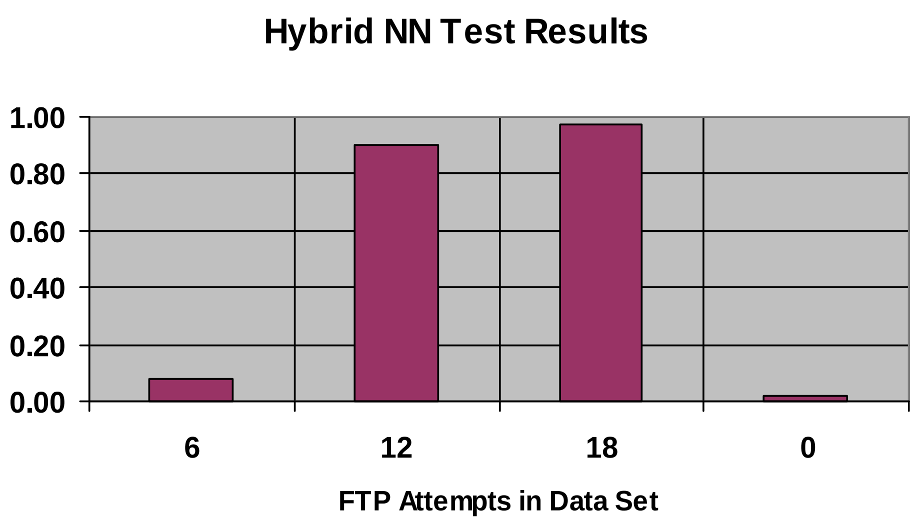
- Designed to test ability of NN to detect:

  1. Temporally dispersed patterns
  2. Collaborative attacks

- Tested using simulated FTP “brute force” attacks
- Hybrid Architecture:
  - Self-Organizing Map
    - classification of events
    - self-organizing NN
    - 25 x 20 map
  - Multi-level Perceptron
    - pattern recognition
    - designed to identify patterns of 12 or more simulated attacks in each 180 event set
  - Trained with 50 data sets containing 1 “attack” interleaved with 50 “normal” data sets


t = 180
h t t p : / / e s e t . s c l s
s u p e r v I s o r
a n o n y m o u s
h t t p : / / w w w . g t l s
c
f t p : / / a s t r o . g a t c
l o g o u t
h t t p : / / e s e t . s c l s
f t p : / / a s t r o . g a t c
t e l n e t c l s . n o v a
s u p e r v I s o r
a d m I n I s t r a t o r
h t t p : / / w w w . c n n . c
h t t p : / / w w w . g t l s c
h t t p : / / w w w . m i c r o
a n o n y m o u s
Network Stream
Raw Data and Destination Port of Network Packets
Self-Organizing Map
Results of SOM Event Classification
Multi-Level Perceptron
Neural Network Output

Sentinel: Prototype #2 Results



- Tested with data sets containing 6, 12, 18, and 0 “attacks” in each 180 event data set
- Successfully detected >= 12 “attacks” in test cases
- Failed to “alert” in lower number of “attacks” (per design)

NN Leaning-Training Requirements

NN and machine learning techniques that require baseline behavior profiles require extensive training.

- Time consuming
- Determines quality of results
- Training in one environment may not map well to another environment
- Over training is a problem for some classes of machine learning

NN Approach: Author's claims

- Prototypes have provided positive indications of the viability of a NN approach
- Experimental NN architectures are not designed for "live" dynamic network environment
- Development of adaptive intelligent systems methodology to improve analytical capabilities of Sentinel
- Experiment with different neural network architectures and related systems
  - Adaptive neural networks
  - Statistical Learning Approaches
- Apply NN approach to more complicated attacks and "live" data stream

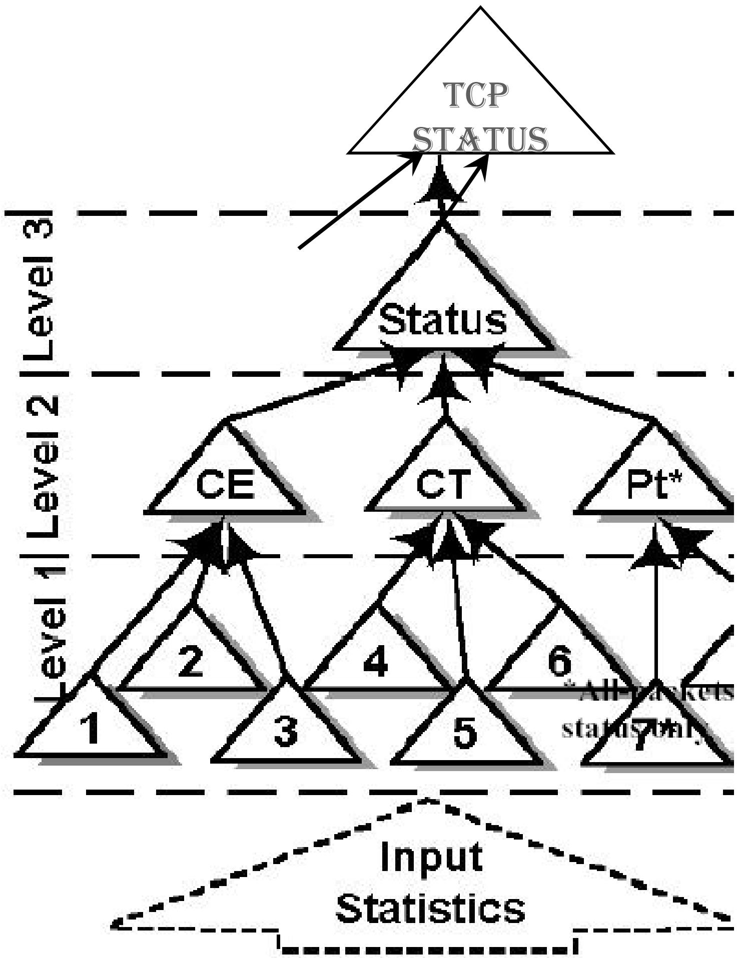
Profiling: NNs for Anomaly Detection

- Build profiles of software behavior and distinguish between normal and malicious software
- Data – strings of BSM (Basic Security Module) events
- Classify entire sessions not single strings of BSM events
- NN with one output node
  - "leaky" bucket algorithm employed
  - leaky bucket algorithm keeps a memory of recent events by incrementing a counter of the neural network's output, while slowly leaking its value
  - If level in the bucket > threshold $\Rightarrow$ generate alarm
  - emphasizes temporal co-located anomalies

(A. Ghosh, A. Schwartzbard, A Study in Using Neural Networks for Anomaly and Misuse Detection 1999.)

Use of NNs for Anomaly Detection

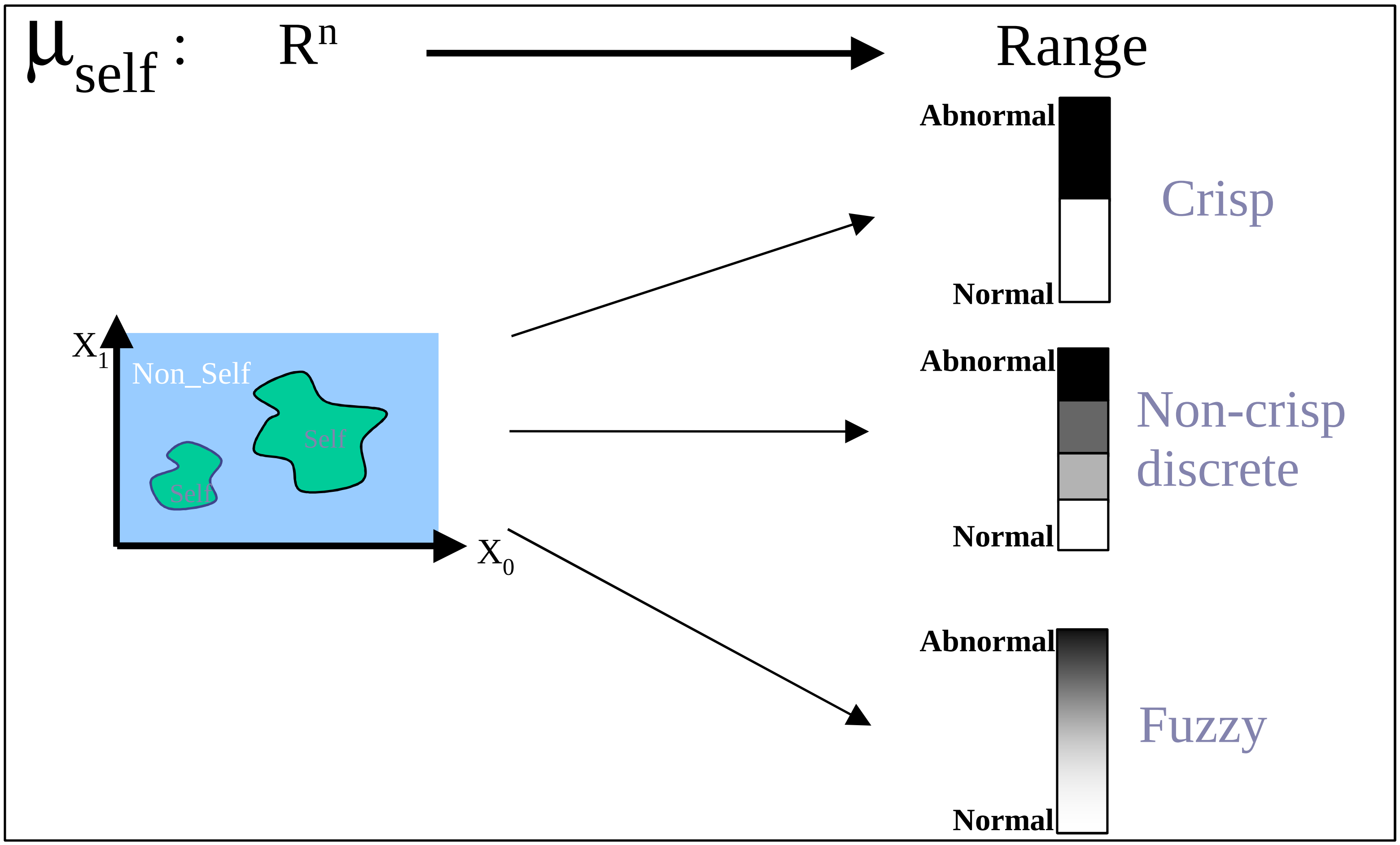
- Three-level architecture
  - Packets and queue statistics are used as inputs to the level 1 NNs
  - The outputs from the Level 1 NNS are combined into:
    - Connection establishment (CE)
    - Connection termination (CT)
    - Port use (*Pt* for all packets only)
  - Outputs from Level 2 are combined at Level 3 into a single status
  - Each of these status monitors are further combined to yield a single TCP status



(S. Lee, D. Heinbuch, Training a Neural-Network Based Intrusion Detector to Recognize Novel

Fuzzy Logic in Cyber Security

Fuzzy Anomaly Detection Function



Fuzzy Logic and Intrusion Detection (ID) Problem (J. Gomez, 2002)

Fuzzy classifier system for solving intrusion detection problem should have a set of $m+1$ rules, one for the normal class and m for the abnormal classes, where the condition part is defined by the monitored parameters and the consequent part is an atomic expression for the classification attribute

Fuzzy Logic in Anomaly-Based ID

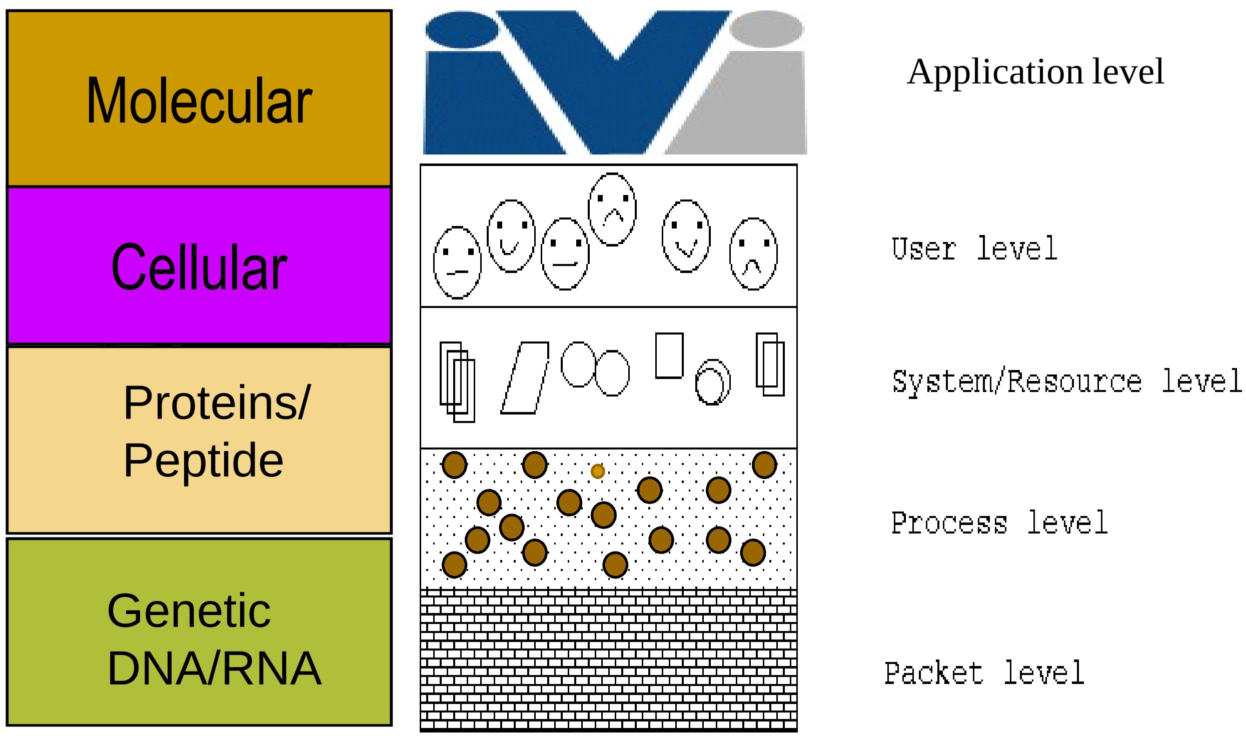
R_{Normal} : *IF x is HIGH and y is LOW*
THEN pattern is normal [0.4]

$R_{Abnormal\text{-}1}$: *IF x is MEDIUM and y is HIGH*
THEN pattern is abnormal$_1$ *[0.6]*

. . .

$R_{Abnormal\text{-}m}$: *IF x is LOW*
THEN pattern is abnormal$_m$ *[0.7]*

Multi-Level Monitoring & Hierarchical Detection Schemes



Multi-Level Monitoring & Correlation

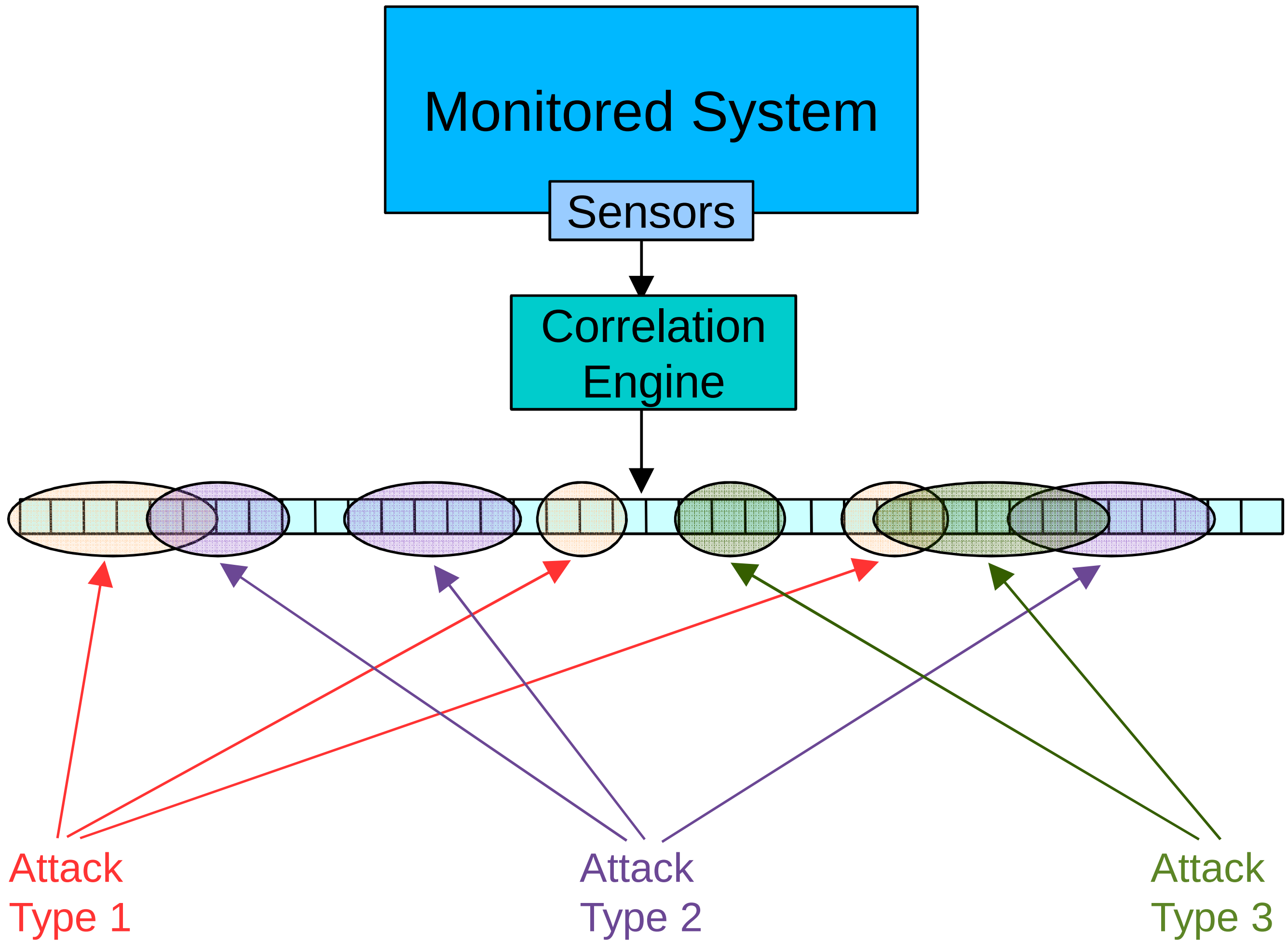


Illustration of Fuzzy rules in ID

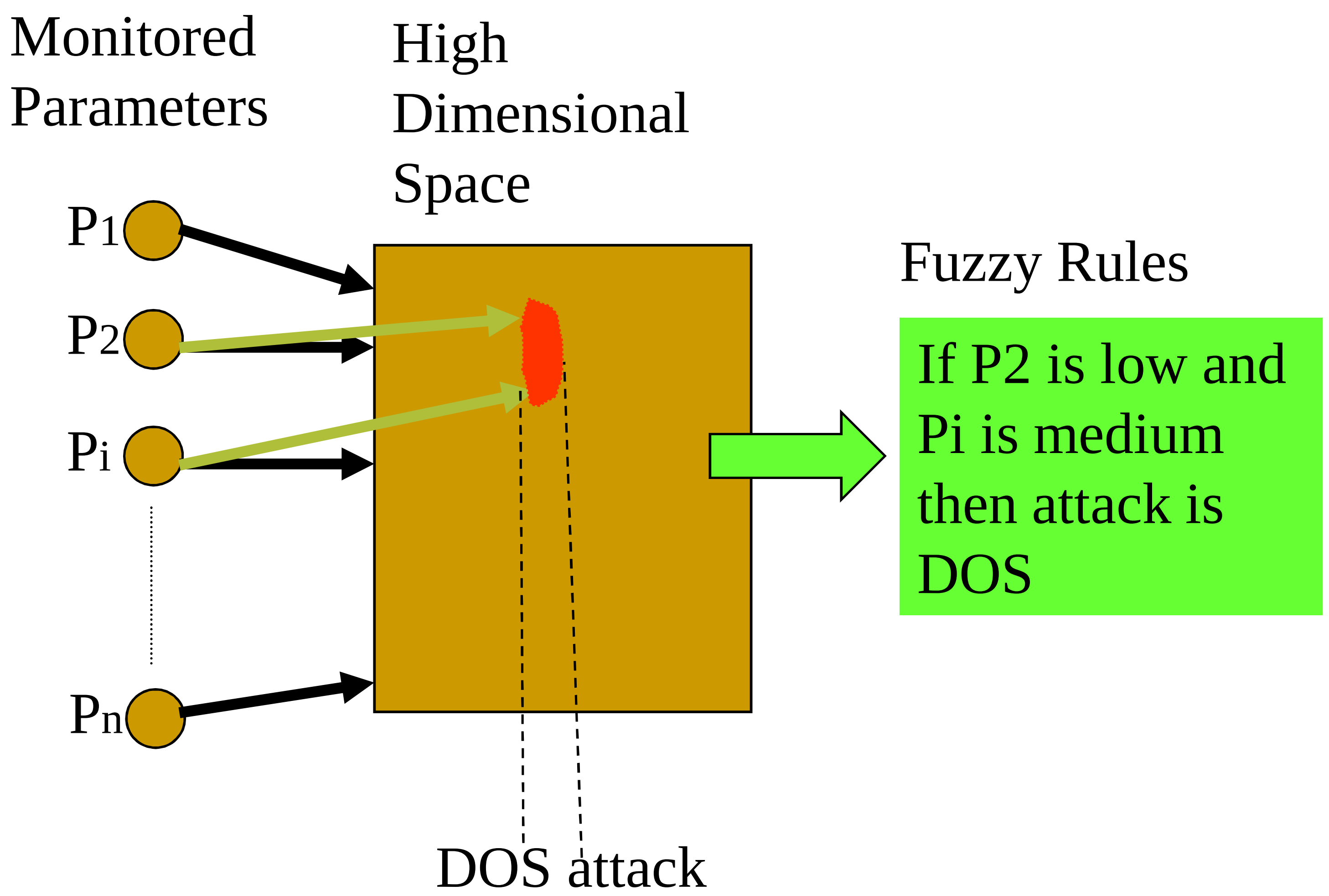
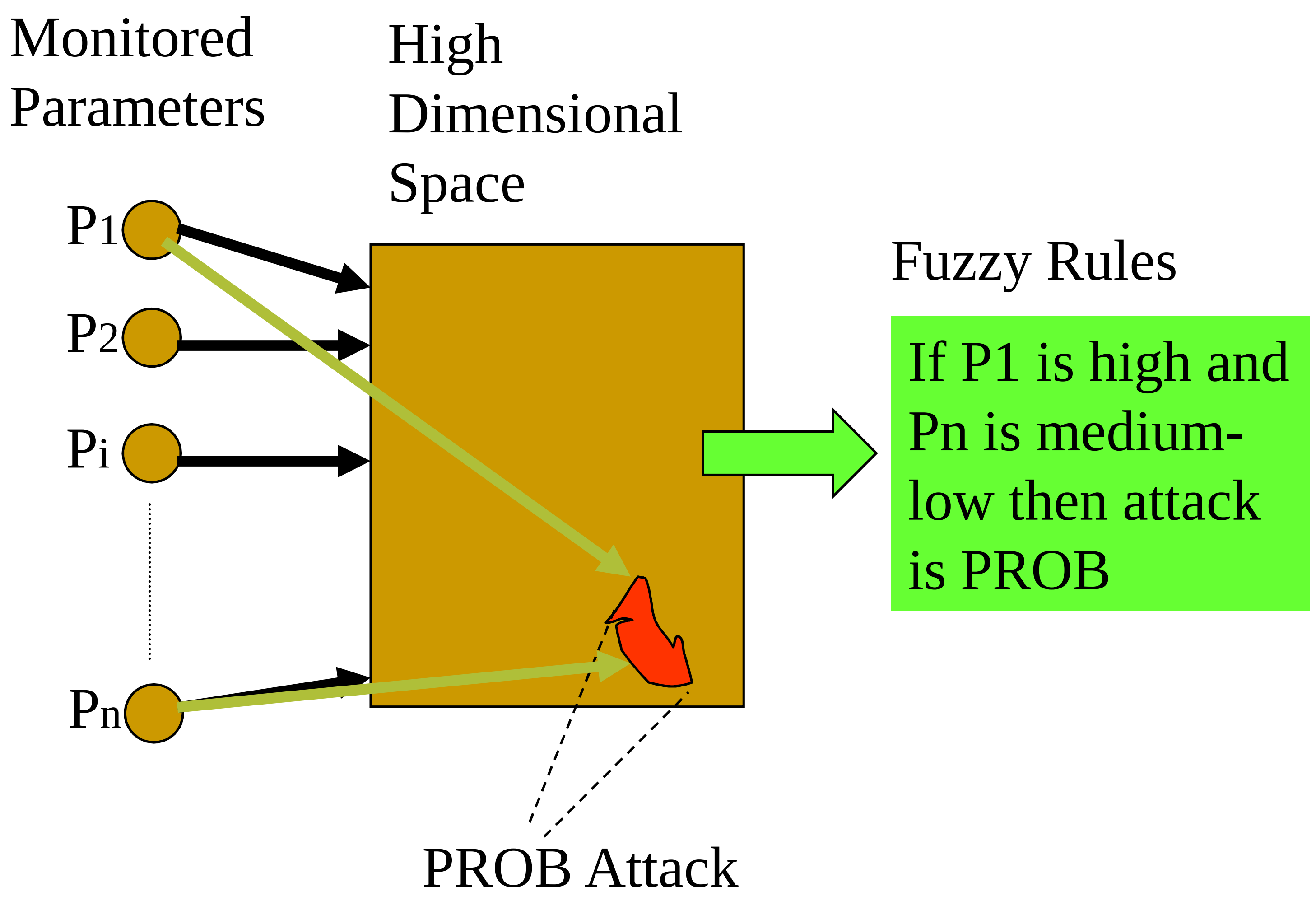


Illustration of Fuzzy rules in ID

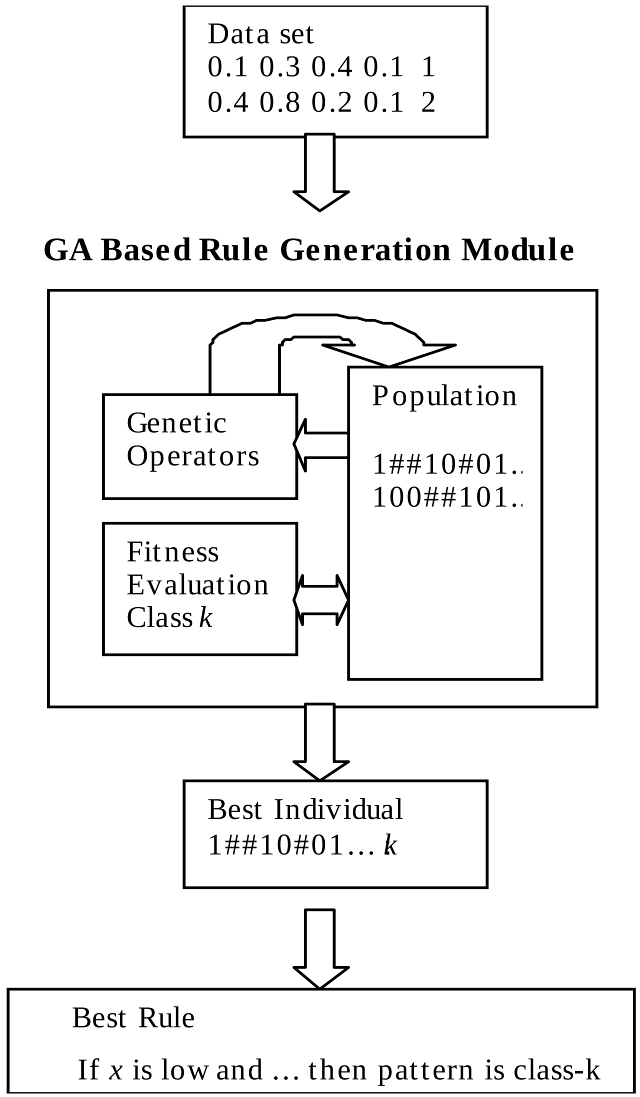


Fuzzy rules for ID

- There are several techniques to generate the fuzzy classifier system for solving the Intrusion Detection System
  - A human expert can write the set of fuzzy rules
  - Fuzzy rules can be extracted from a neural network that solves the problem
- Gomez et. al. 2002 developed Genetic Algorithm based rule generators.

Evolving Fuzzy Rule for ID

Steps to produce a fuzzy rule for the class *k* attack types using a Genetic Algorithms



Fuzzy Rule Evolution: *Fitness measure*

$$TP = \sum_{i=1}^{p} predicted\ (class\_data_i)$$

$$TN = \sum_{i=1}^{q} [1 - predicted\ (other\_class\_data_i)]$$

$$FP = \sum_{i=1}^{q} predicted\ (other\_class\_data_i)$$

$$FN = \sum_{i=1}^{p} [1 - predicted\ (class\_data_i)]$$

$$sensitivity = \frac{TP}{TP + FN}$$

$$specificity = \frac{TN}{TN + FP}$$

$$length = 1 - \frac{chrom\_length}{10}$$

$$fitness = w_1 * sensitivity + w_2 * specificity + w_3 * length$$

Intrusion Data sets (DARPA)

- Network data obtained from the MIT-Lincoln Lab (*tcpdump*).
- The data represents both normal and abnormal information collected in a test network.
- For each TCP/IP connection, 41 various quantitative and qualitative features were extracted
- It contains complete weeks with normal data. This allowed us to get enough samples to build the training dataset.

DARPA Data (Attack Classes)

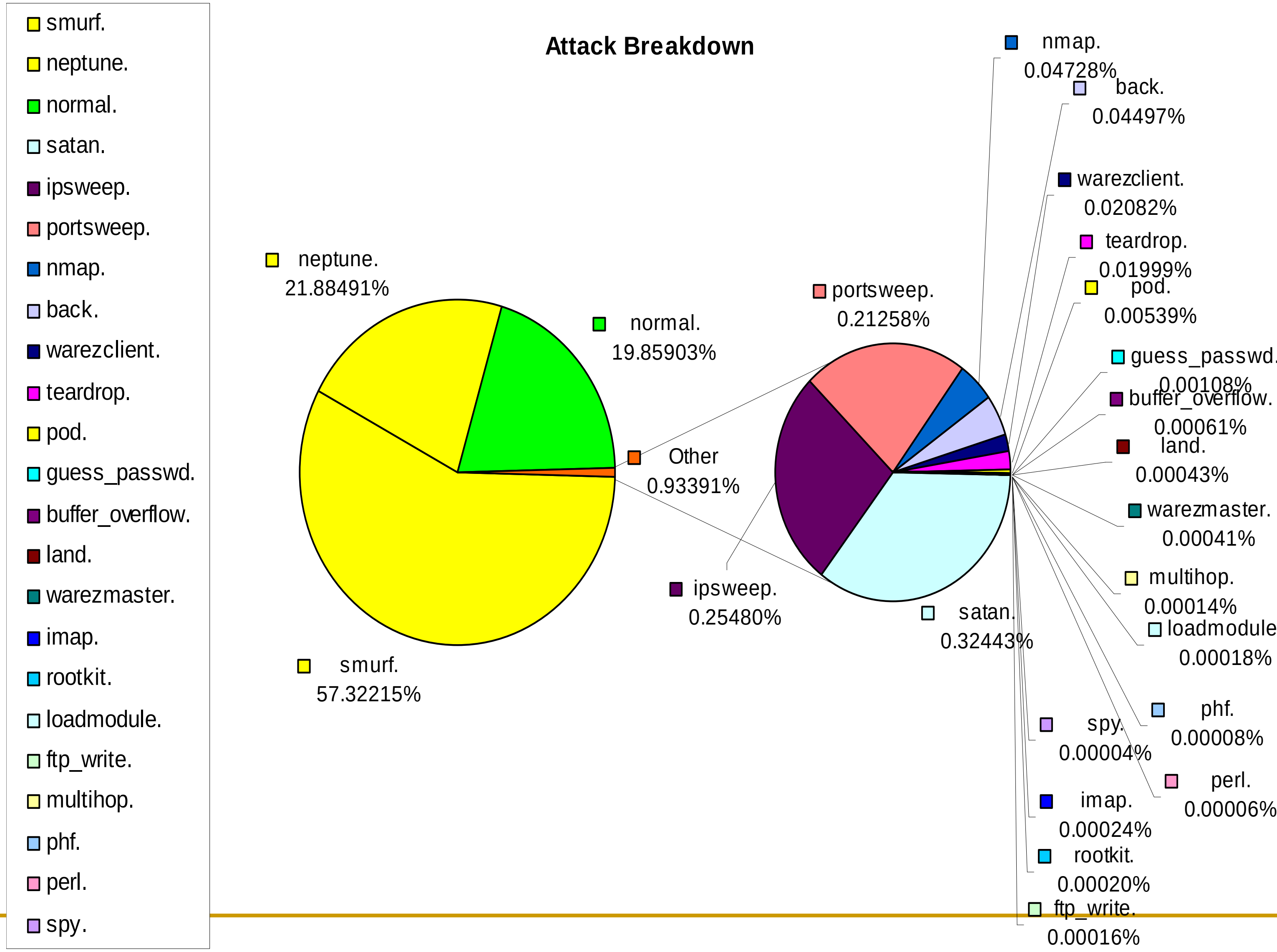
Attacks fall into four main classes:

- Probing: surveillance and other probing.
- DOS: denial of service.
- U2R: unauthorized access to local super user to (root) privileges.
- R2L: unauthorized access from a remote machine.

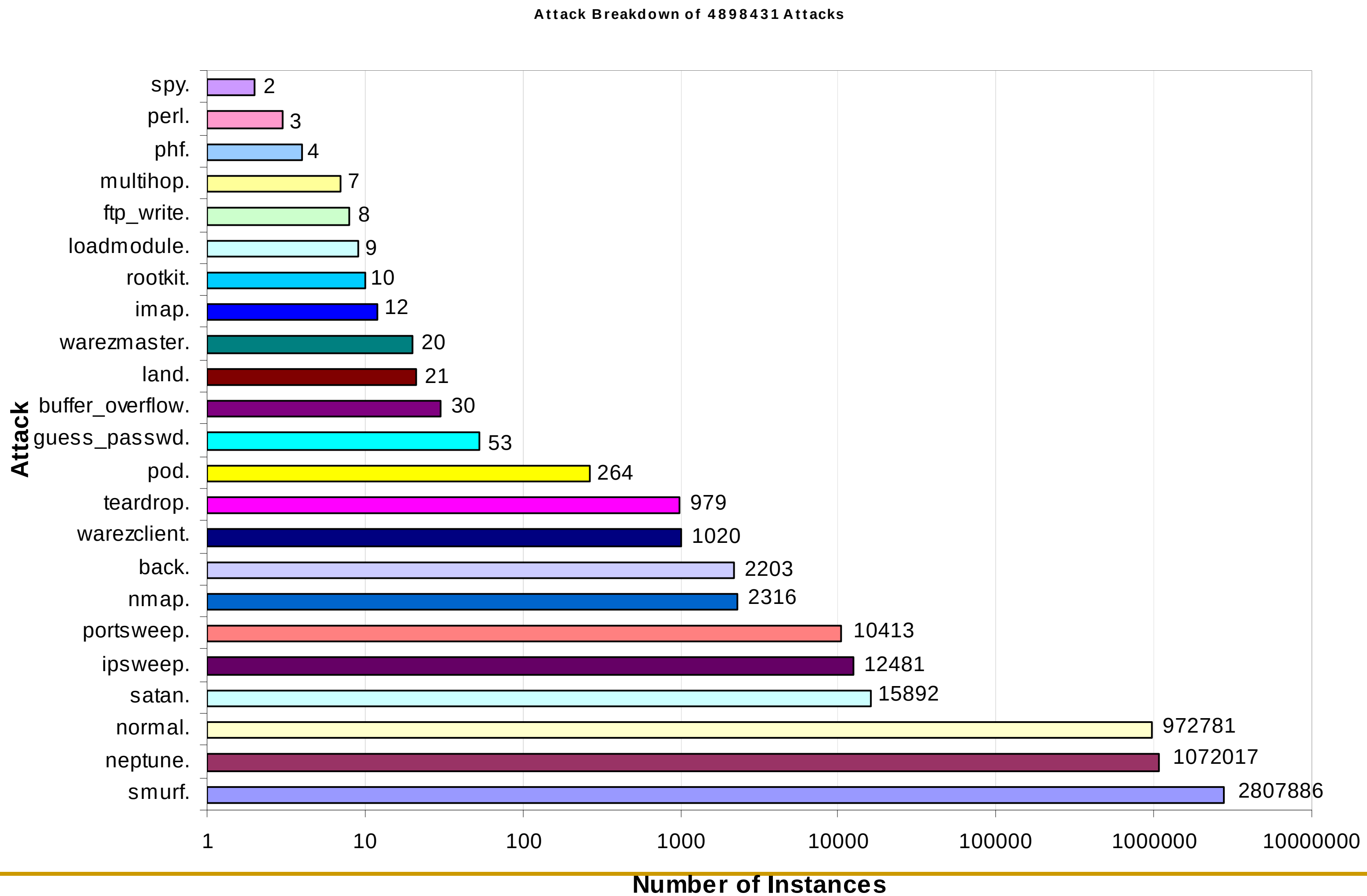
DARPA Test Dataset: *Details*

CLASS	SUB-CLASSES	SAMPLES
Normal		95278 (19.3%)
U2R	buffer_overflow, loadmodule, multihop, perl, rootkit	59 (0.01%)
R2L	ftp_write, guess_passwd, imap, phf, spy, warezclient, warezmaster	1119 (0.23%)
DOS	back, land, Neptune, pod, smurf, teardrop	391458(79.5%)
PRB	ipsweep, nmap, portsweep, satan	4107 (0.83%)

Analysis of DARPA Dataset



Analysis of DARPA data set (cont..)



Fuzzy ID Rules : *Experiments & Results*

The proposed approach was able to generate fuzzy rules (the longest fuzzy rule contains only five atomic expression). The following are some fuzzy rules that were evolved in a sample run:

if (dst_host_srv_count is not low or protocol_type is not tcp) and protocol_type is not icmp **then** record_type is normal [1.0]

if dst_host_srv_count is low and flag is not S0 and protocol_type is not icmp and dst_host_srv_rerror_rate is not level-4 **then** record_type is U2R [1.0]

if (dst_host_srv_count is low or is_guest_login is true) and flag is not REJ and dst_host_same_srv_rate is not low and duration is not level-4 **then** record_type is R2L [1.0]

if count is not low or same_srv_rate is low **then** record_type is DOS [1.0]

if dst_host_same_srv_rate is low and flag is not SF or protocol_type is icmp

then record_type is PRB [1.0]

Fuzzy ID Rules : *Comparative Results*

Performance reached by the Fuzzy approach and some methods reported in the literature. Here

FA: False alarm rate
DR: Detected attacks rate

Algorithm	FA %	DR %	Complexity
EFRID	**7.0**	**98.95**	**O(n)**
RIPPER-Artificial Anomalies	2.02	94.26	$O(n*\log^2 n)$
SMARTSIFTER	-	82.0	$O(n^2)$

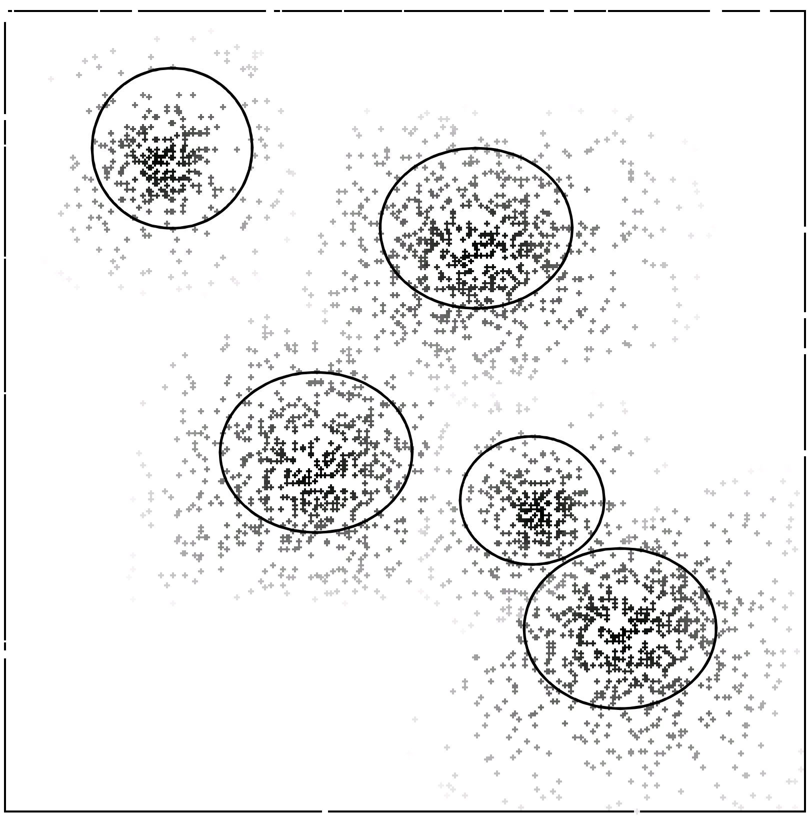
Gravitational Clustering in Intrusion Detection (J. Gomez 2003)

- A set of collected normal data records defines the training data set
- The basic ideas behind applying the gravitational law are:
  1. A data point in some cluster exerts a higher gravitational force on a data point in the same cluster than on a data point that is not in the neighborhood.
  2. If some points are noisy (out-layer), the gravitational force exerted on them from other points are very small.

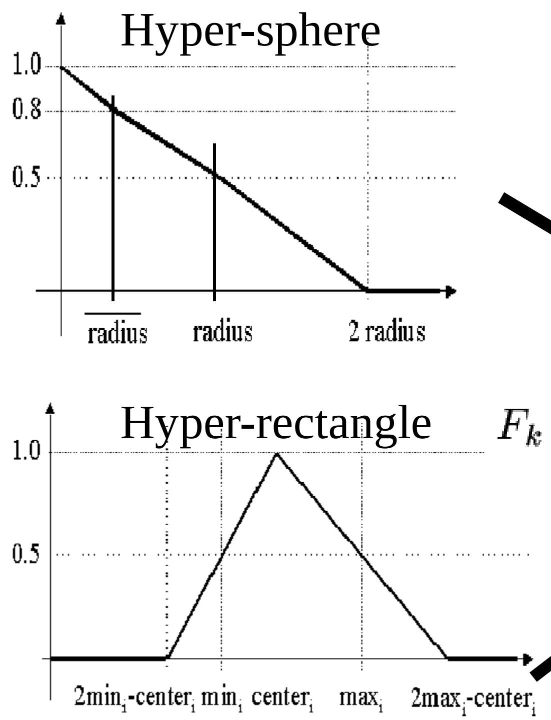
Fuzzy Gravitational Clustering

1. Generate a set of clusters (with the gravitational clustering) that represents the normal behavior (Positive Characterization).
2. Assign data points to the closest cluster
3. Calculate statistical information such as min, max, radius, avg radius, etc.
4. Generate a fuzzy membership function for the generated clusters using such statistical information

Fuzzy Gravitational Clustering in ID



Gravitational Clustering



$$F_k(x) = \frac{2 * f_k(x) * g_k(x)}{f_k(x) + g_k(x)}$$

$$g_k(x) = \min_{1 \le i \le n} \{g_{k,i}(x_i)\}$$

Summary of Gravitational Clustering in ID

- The applicability of the gravitational clustering algorithm and the fuzzy cluster analysis, in solving some well studied intrusion detection problems.
- Gravitational clustering algorithm generates a good set of clusters for characterizing the normal behavior by using only the normal training samples.
- The fuzzy cluster analysis performed over the clusters generated pays off in the characterization of the boundaries between normal and abnormal spaces.
- Experiments showed that the performance of the proposed approach is comparable with other results reported in the literature.

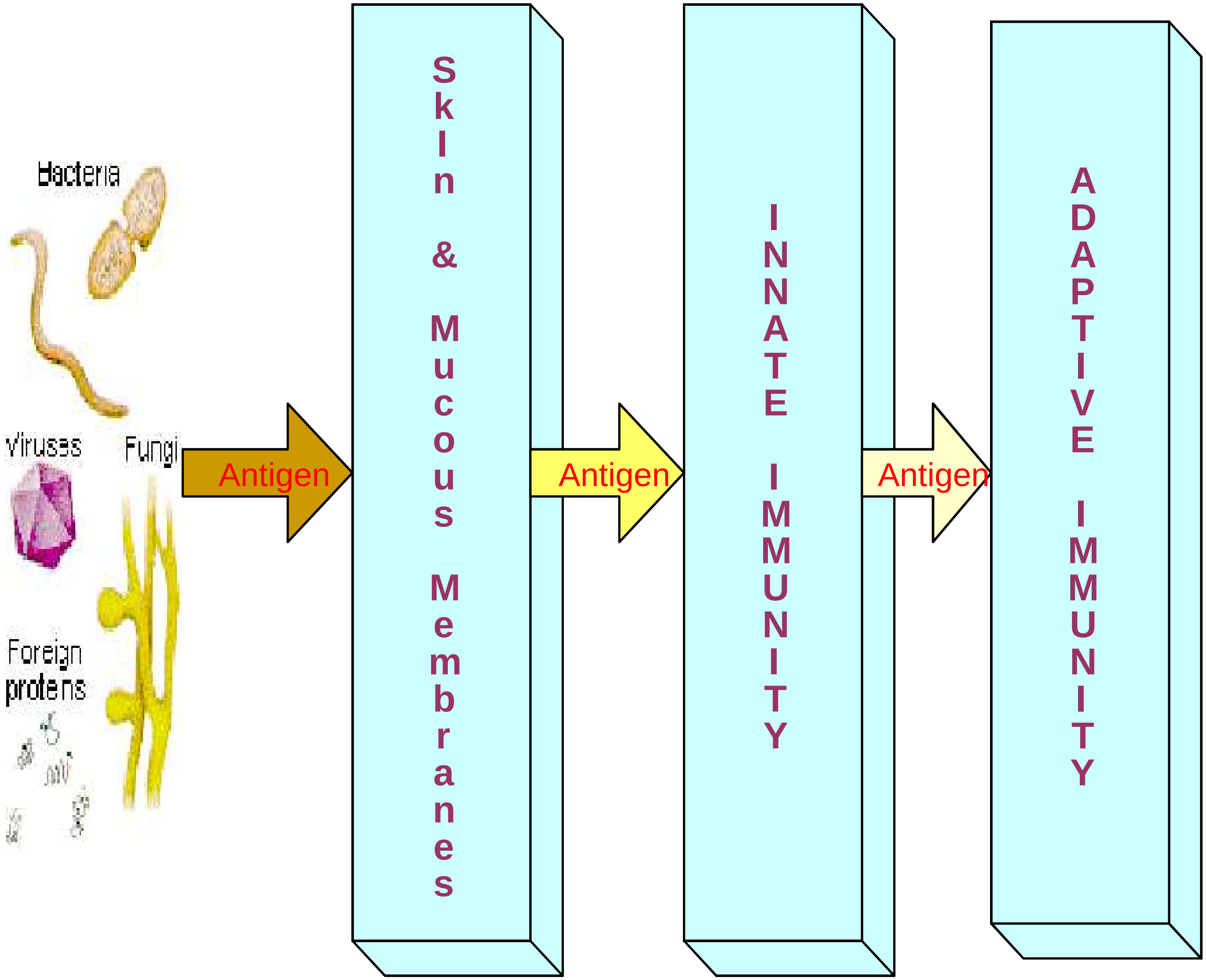
Immunity-Based Approaches

in Cyber Security

The Biological Immune System – A Defense System **(S. Forrest 1994)**

- Its primary role is to distinguish the host (body cells) from external entities (pathogens).
- When an entity is recognized as non-self (or dangerous) - activates several defense mechanisms leading to its destruction (or neutralization).
- Subsequent exposure to similar entity results in rapid immune response (Secondary Response).
- Overall behavior of the immune system is an emergent property of many local interactions.

Multiple levels of protections



Increased Complexity

Skin and Mucous Membrane

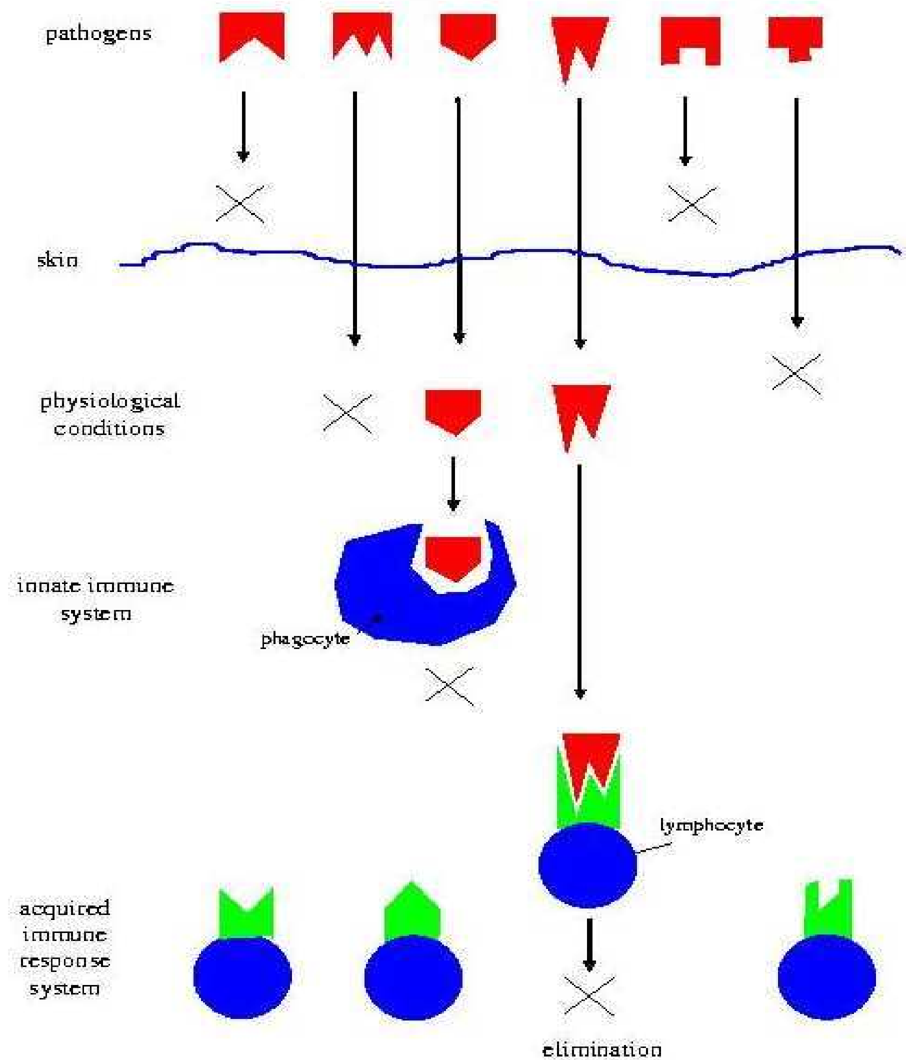
- The skin outer layer consists of dead cells and is filled with a waterproofing protein, which can prevent the penetration of most pathogens.
- Some glands in the skin inner layer can produce low PH oily secretion, which inhibits the growth of most bacteria.
- In mucous membranes, saliva, tears, and mucous secretions act to wash away potential invaders and also contain antibacterial or antiviral substances. In the lower respiratory tract and the gastrointestinal tract, the mucous membrane is covered by *cilia*, hairlike processes projecting from the epithelial cells. The synchronous movement of cilia propels mucous-entrapped microorganisms from these tracts .

Innate and Adaptive immunity

- **Innate immunity is nonspecific and handles most of the pathogens. Innate immunity is present at birth, does not develop memory.**

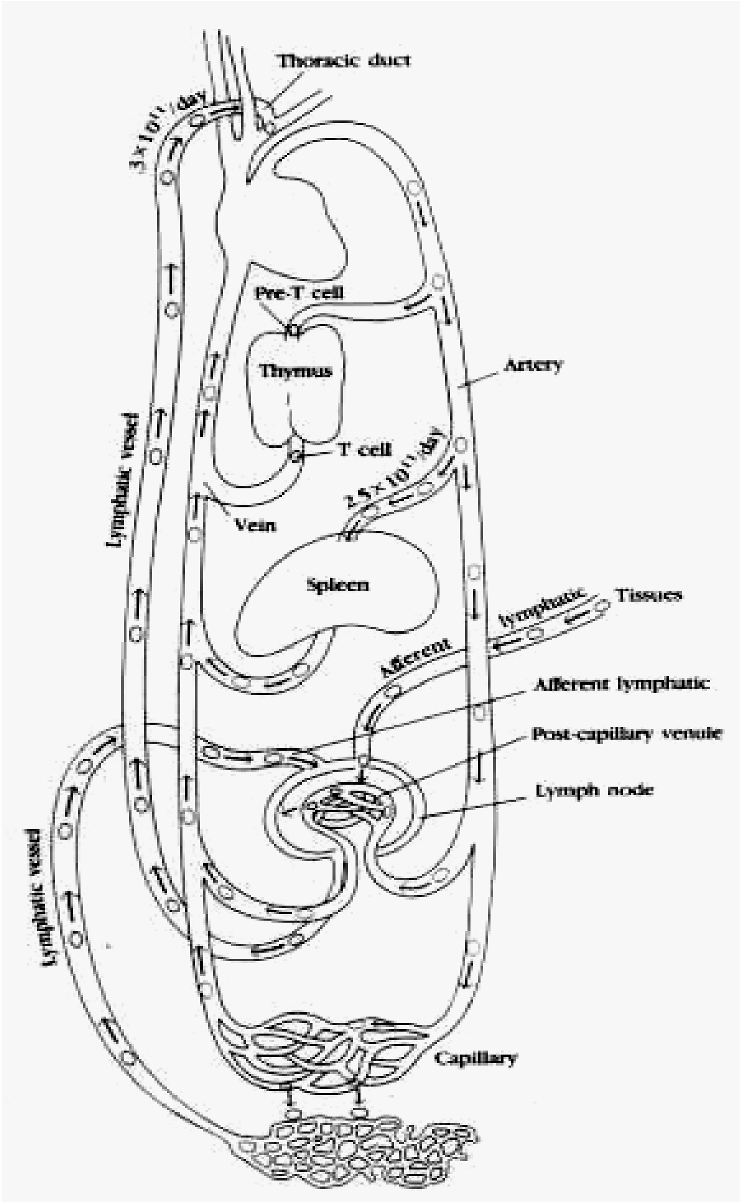
- **Adaptive immunity is specific and has the hallmarks of learning, adaptability, and memory, it is divided into two branches: humoral and cellular immunity.**

Illustration of Multi-Level Protection (Hofmeyr' 96)



Humoral & Cellular immunity

- **Humoral immunity is mediated by antibodies contained in body fluids (known as humors). It involves interaction of B cells with antigen and their subsequent proliferation and differentiation into antibody-secreting plasma cells.**
- **Cellular immunity is cell-mediated; It plays an important role in the killing of virus-infected cells and tumor cells. Cytokines are the key to cellular immunity.**



Circulatory mechanism

(Kuby'94)

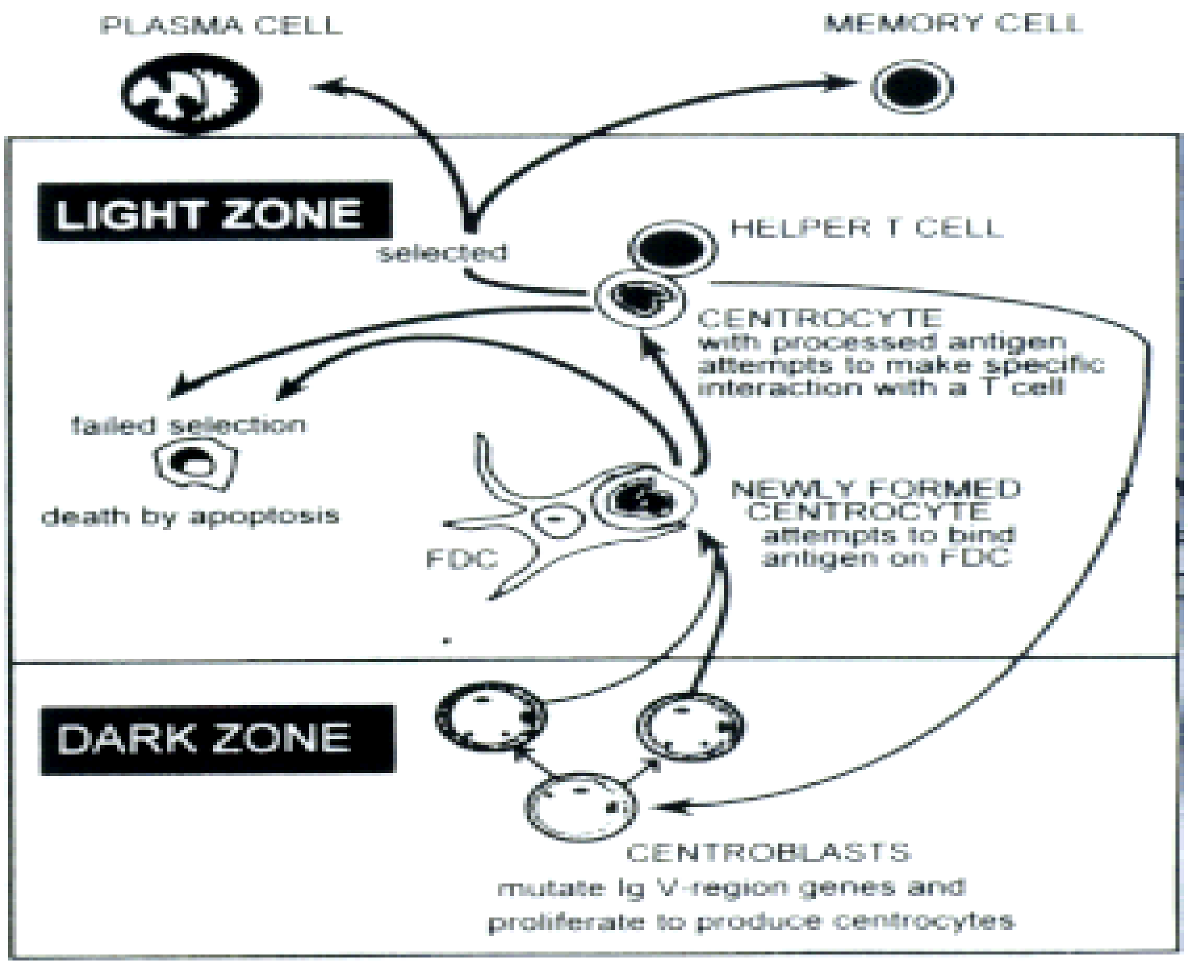
Immune cells circulates constantly through the blood, lymph, lymphoid organs and tissue spaces. They visit primary and secondary lymphoid organs to interact with foreign antigens.

Germinal Center

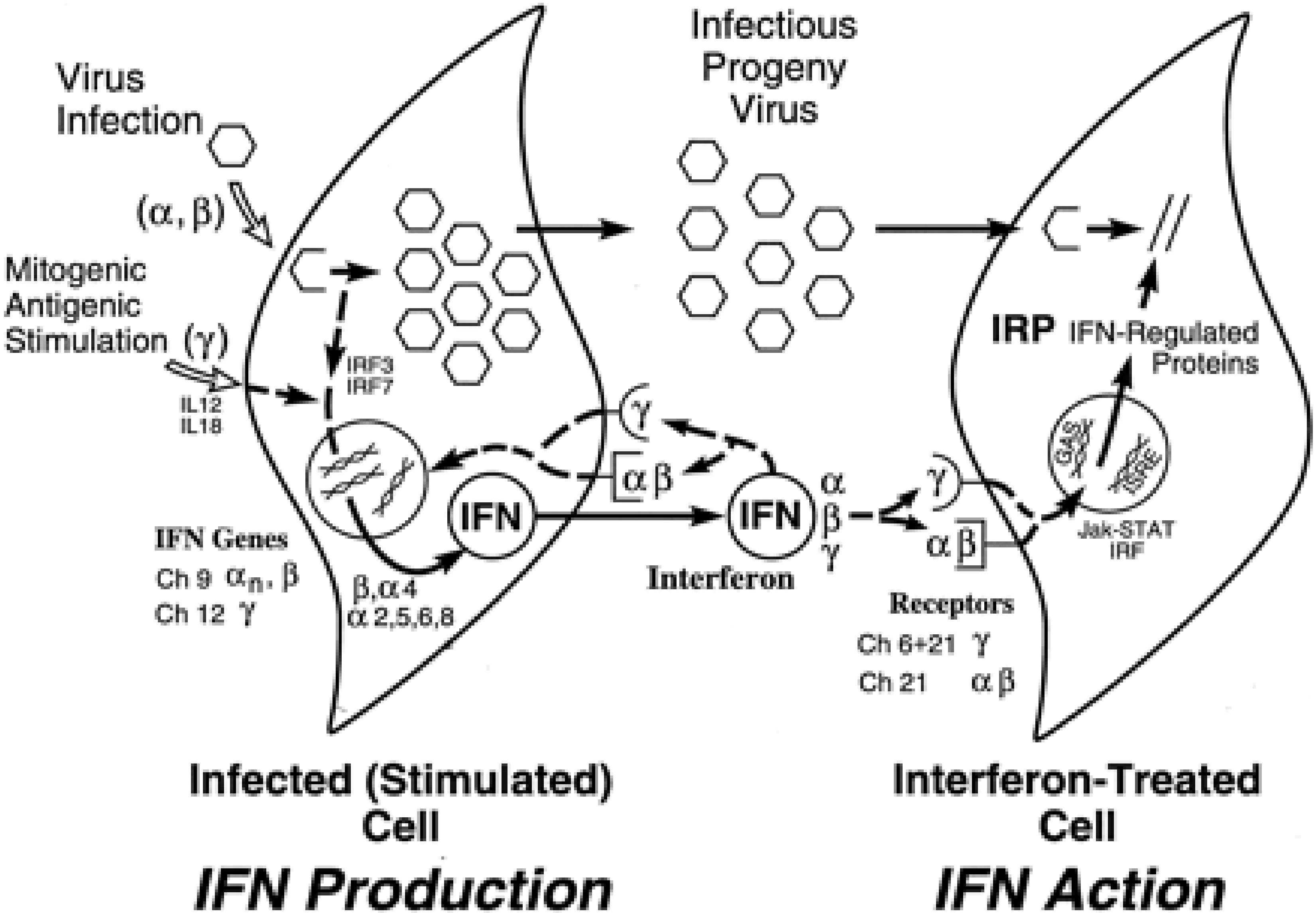
Germinal center is a dynamically evolved structure (in secondary lymphoid organs) which *develops through a complex immunogenetic process and* provides a specialized micro-environment in order to perform many critical functions during some antigenic immune responses.

It work as a mobile Forensic Lab

Mechanisms of GC reaction (Gulbranson-Judge et al. 98)



Interferon (*IFN*) Signaling Mechanisms

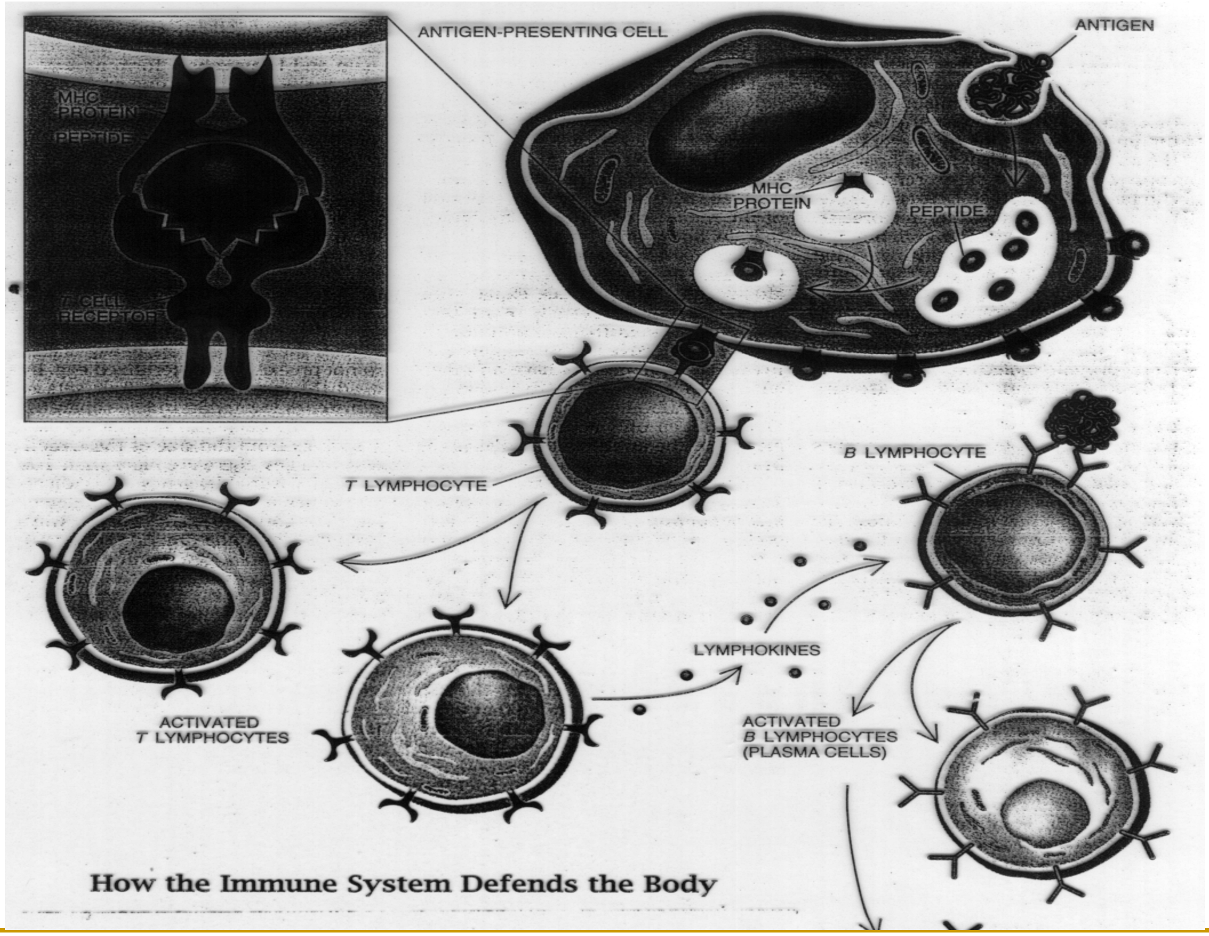


The Goal of Signaling

- **To move a signal from outside the cell to the inside.**
- **This signaling results in changes to the cell, allowing the cell to appropriately respond to the stimulus.**
- **This process of cellular communication results in:**
  - **Surface marker changes**
  - **Changes in cellular distribution**
  - **Environmental changes**
  - **Destruction of foreign invaders**
  - **Destruction of aberrant cells**

Feature Extraction & Co-stimulation mechanisms

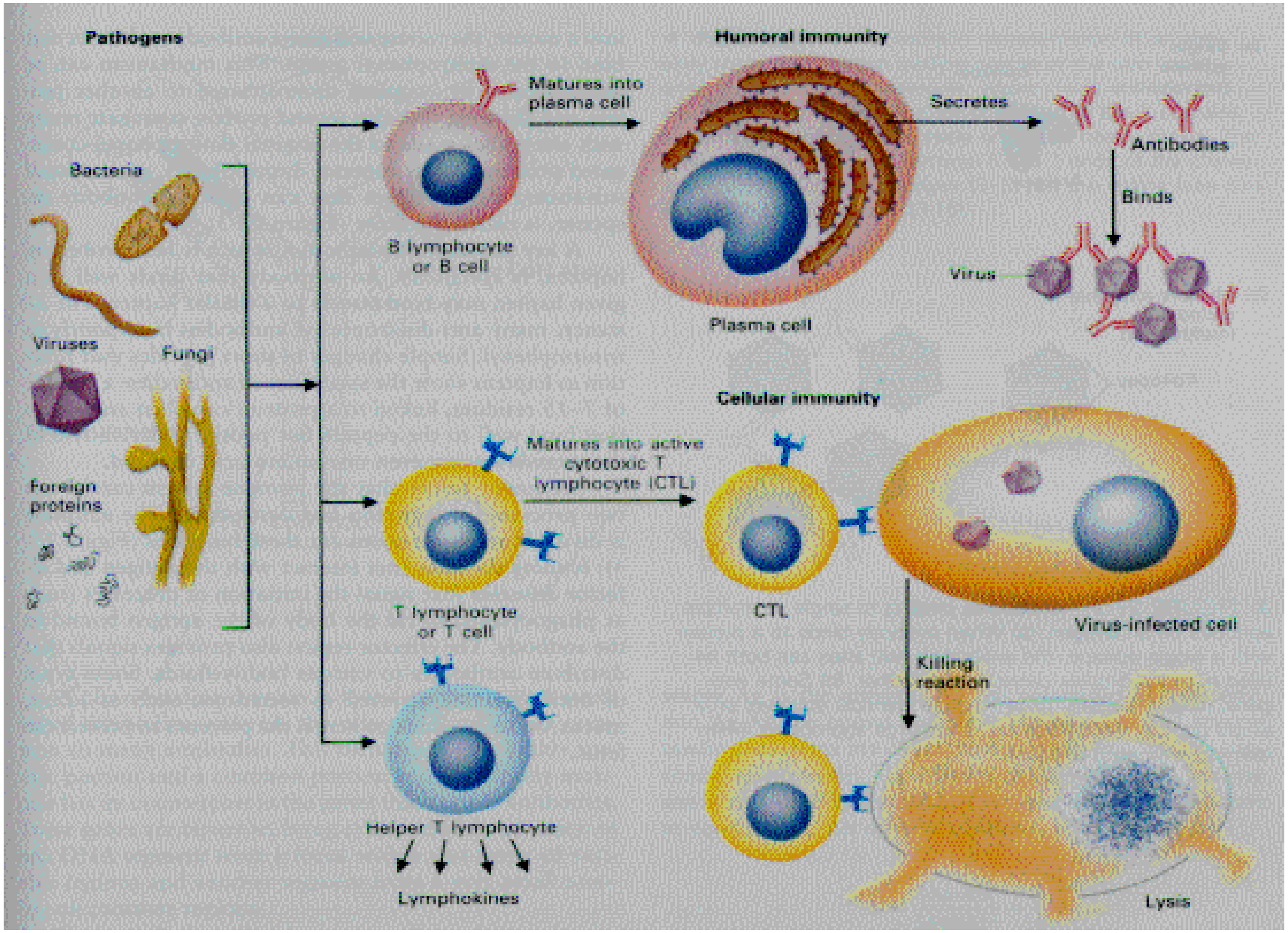
- (Scientific American, Sept. 93)



Protective Immunity

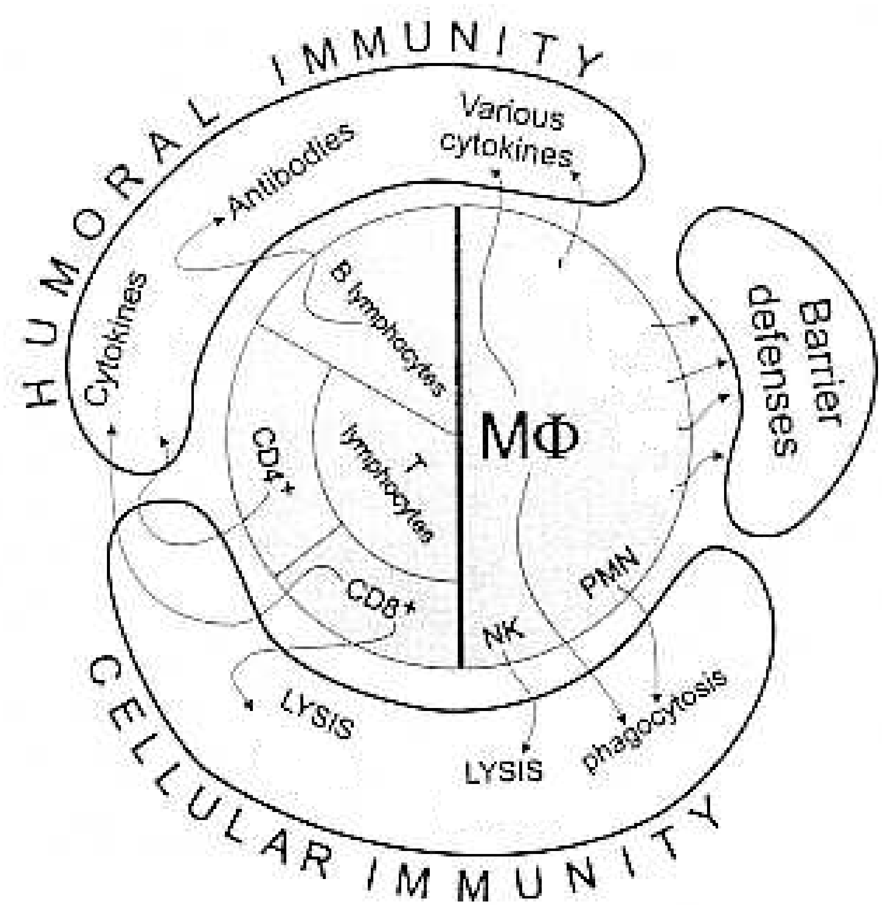
The four players involved in protective immunity—plasma cells, memory B cells, effector T cells, and memory T cells—differ in the longevity of their responses, have different maintenance requirements, and act in different ways to confer protection.

Differential Response Pathway



Overall Immunity (Coverage)

with different defense mechanisms (Whitton'98)



NK: natural killer

PMN: polymorphonuclear leukocytes

MΦ: macrophages

IFN: interferon

Immune response is self-regulatory in nature. Its' response action follow one of the branches-- Humoral or Cellular. It also assures steady-state levels of each cell types by cell-division and programmed death.

From the computational point of view, the immune system is a

- Distributed information processing system
- Novel pattern recognizer: Self/non-self (Danger) Discrimination
- Multi-level Defense System
- Having unique mechanisms for
  - Decentralized control
  - Signaling and Message-passing
  - Co-stimulation
  - Learning and memory

Computer Immune Systems

- Negative-Selection Algorithm (Forrest'94)
  - Virus Detection (1994)
  - Unix process monitoring (1996)
  - Network-based Intrusion Detection (1998, 2001)
- Alternative approaches to Virus Detection
  - Decoy Programs (Kephart'94)
  - Self-Adaptive Virus Immune System (Lamont'98'01)
- Immune Agent Architecture (Dasgupta'99)
  - SANTA: Mobile Security Agents (2001)
  - CIDS: A Security Agent Architecture (2002)
- Other works

Virus Detection (Kephart 94)

- Kephart (1994), generated a set of antibodies to previously not encountered computer viruses or worms.
- A particular virus was recognized via an exact or fuzzy match to a relatively short sequence of bytes occurring in the virus (called a “signature”).
- The process by which the proposed computer immune system established whether new software contained a virus had several stages to avoid autoimmune responses.
- Integrity monitors, which used checksums to check for any changes to programs and data files, had a notion of *self* that was: any differences between the original and current versions of any file were flagged, as were any new program.

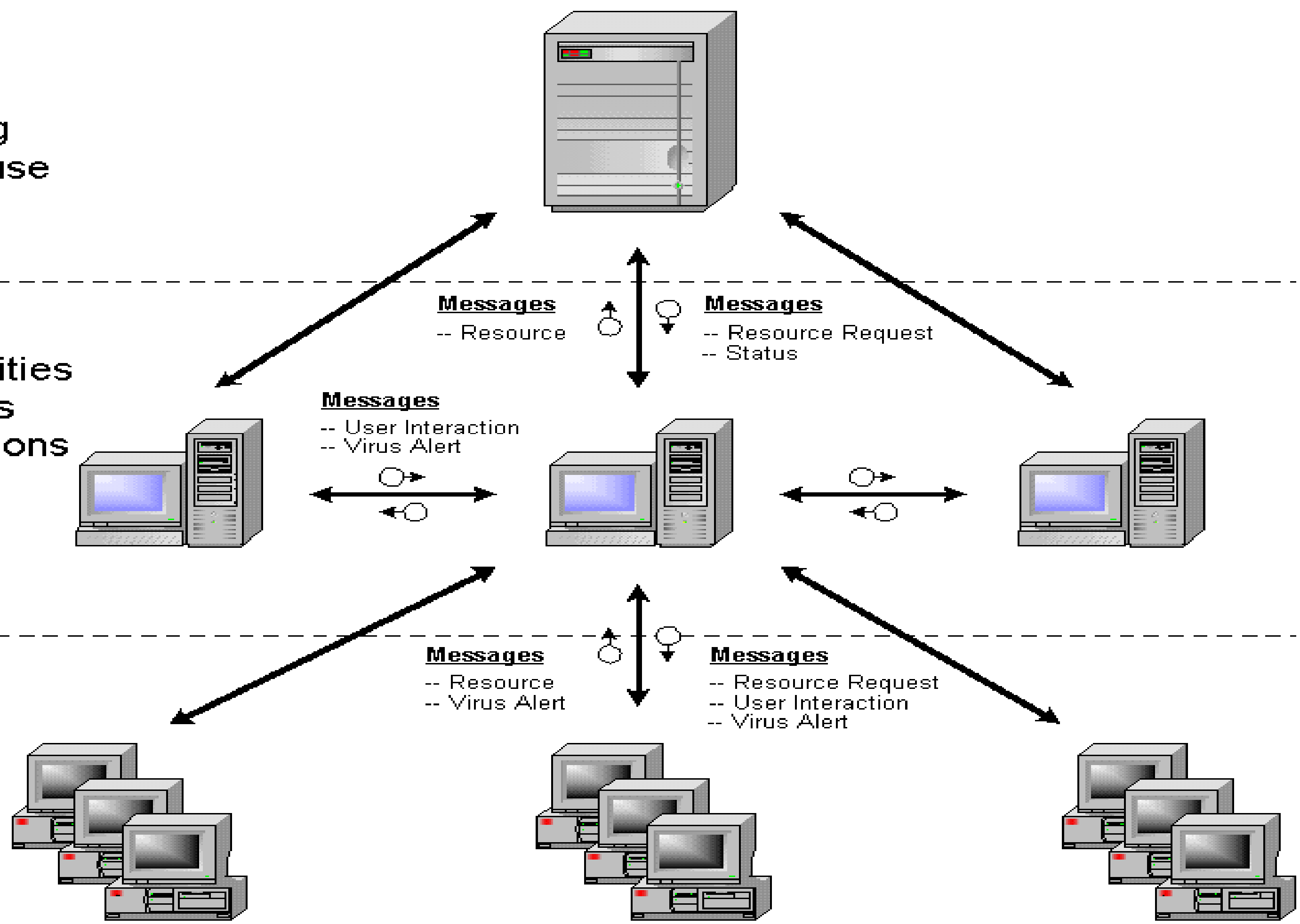
Multi-Level Model for Virus Detection (Lamont'98)

System Model

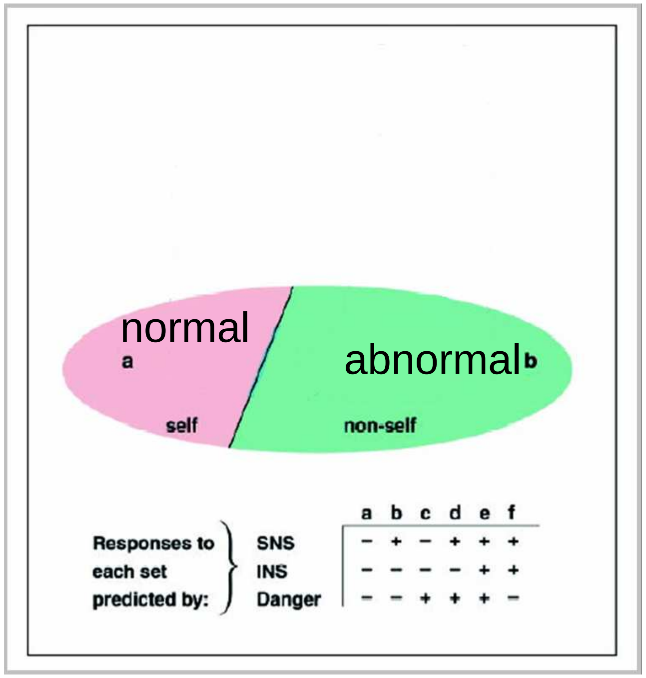
System Level
-- Status Collection
-- Metric Generation
-- Information Sharing
-- Resource Warehouse

Network Level
-- Control Local Activities
-- Collect Local Status
-- Dispense Vaccinations
-- User Interface

Local Level
-- Virus Detection
-- System Response
-- System Memory



Self/Non-Self Model



Partition of the Universe of Antigens

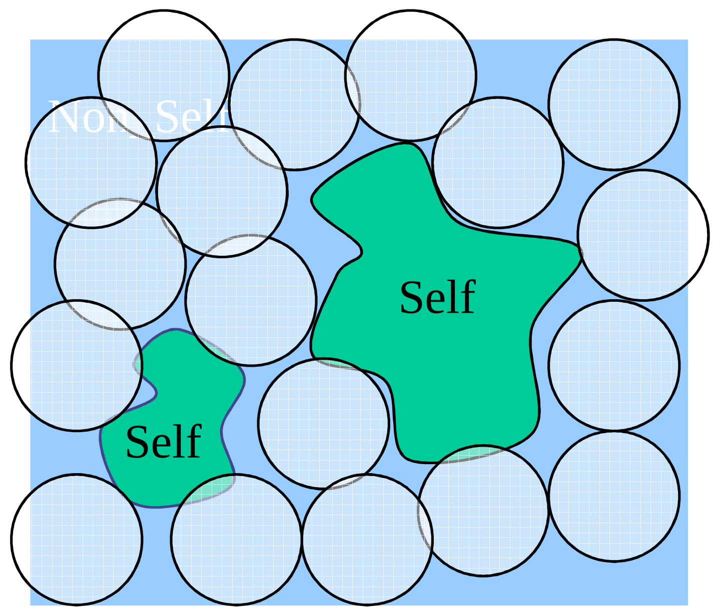
SNS:

self and nonself (*a and b*)

Negative Selection Algorithm (Forrest'94)

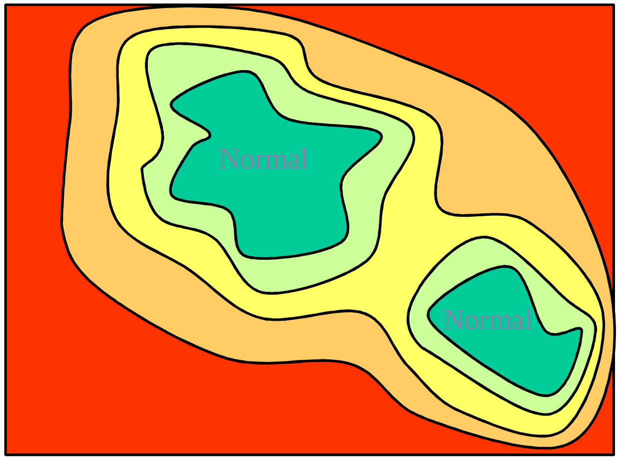
Given self (as a collection of positive samples), generate Points (rules) that can cover the non-self space efficiently.

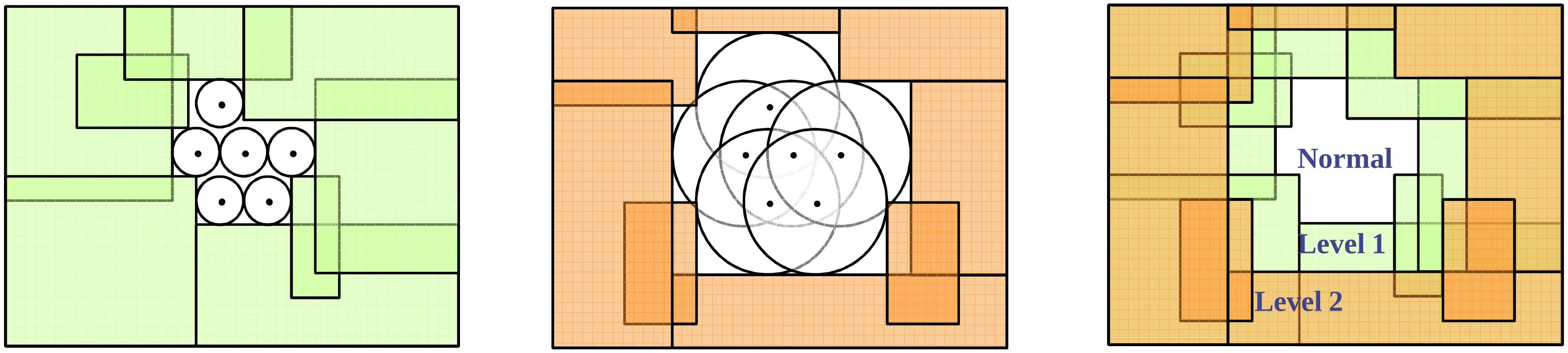
- There exist efficient algorithms that runs on linear time with the size of self (for binary representation).
- Efficient algorithm to count number of holes.
- Theoretical analysis based on Information Theory.



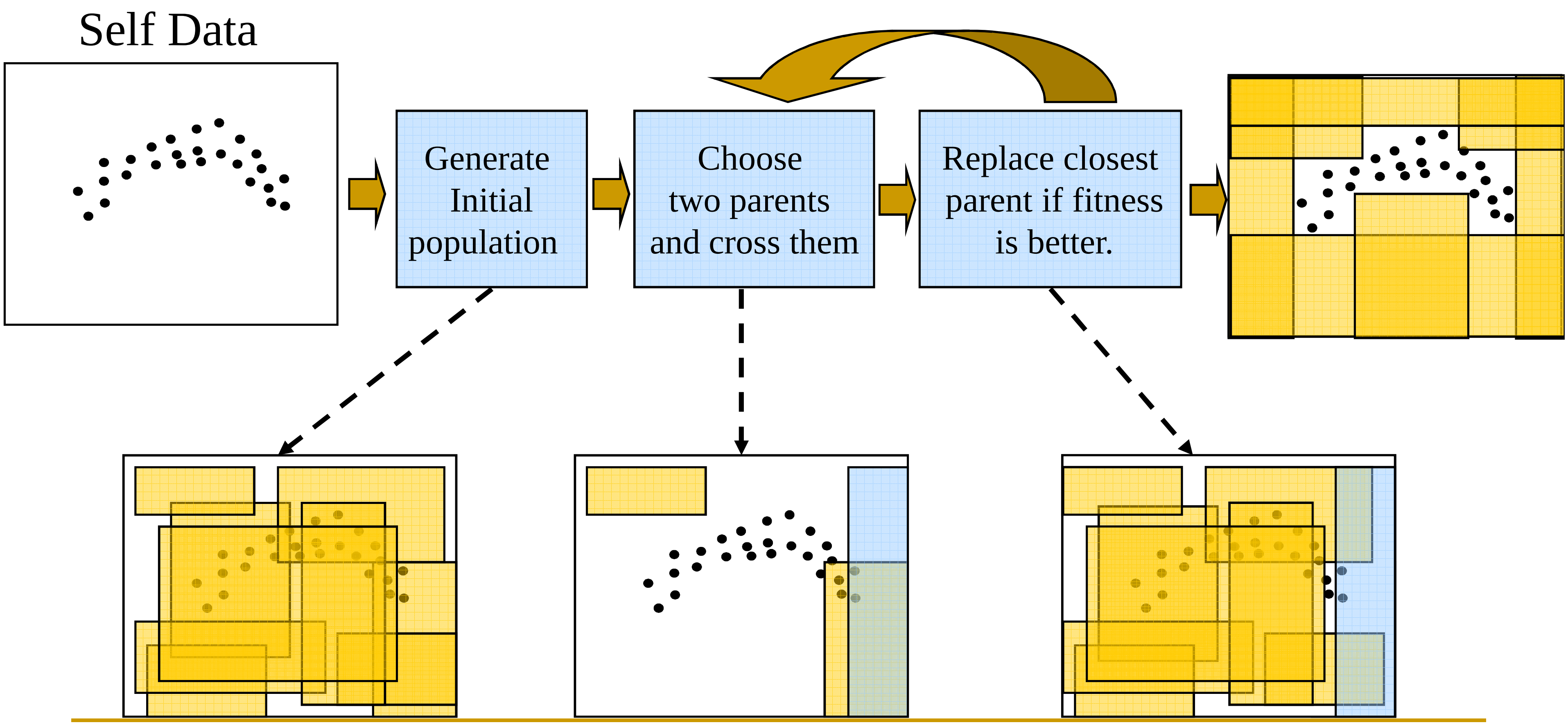
Real-Valued NS (RNS) Algorithm:

- Define different levels of variability on the self set.
- Evolve detectors for the different levels.





RNS Rule Evolution: Block Diagram



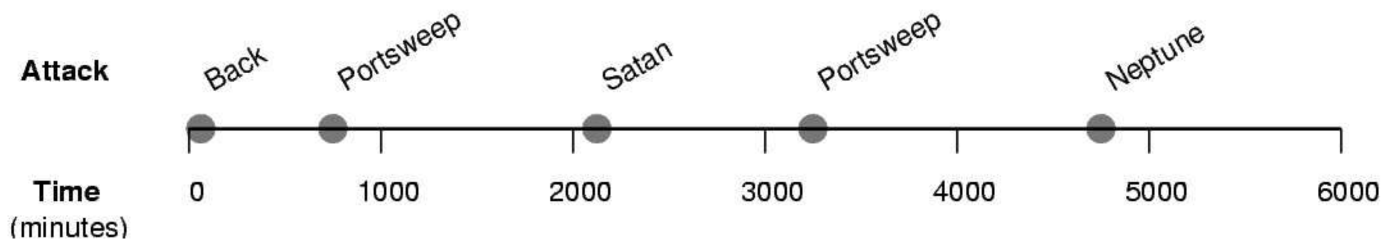
Features used from the Intrusion Data

In our experiments, MIT data was processed to extract the following features:

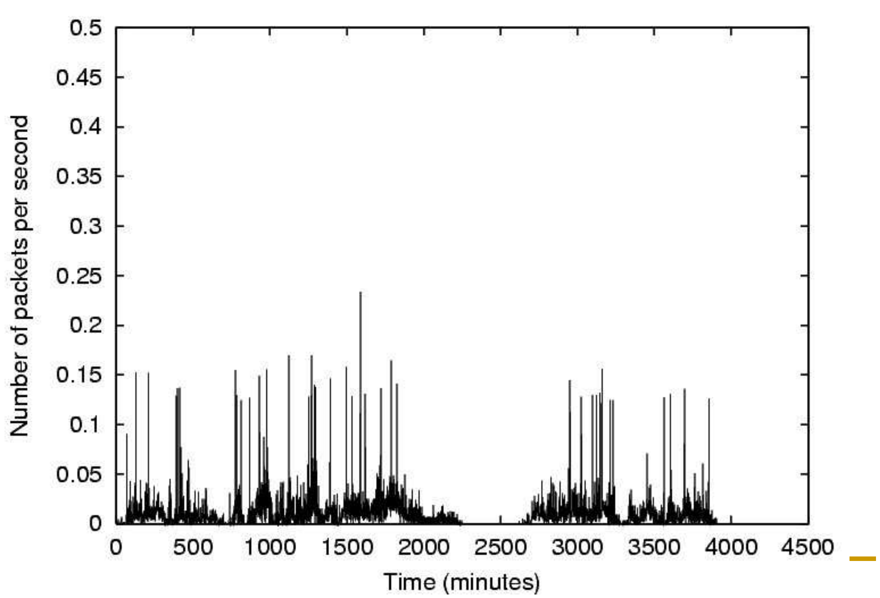
- Number of bytes per second
- Number of packets per second
- Number of ICMP packets per second

Attack Time Line in data sets

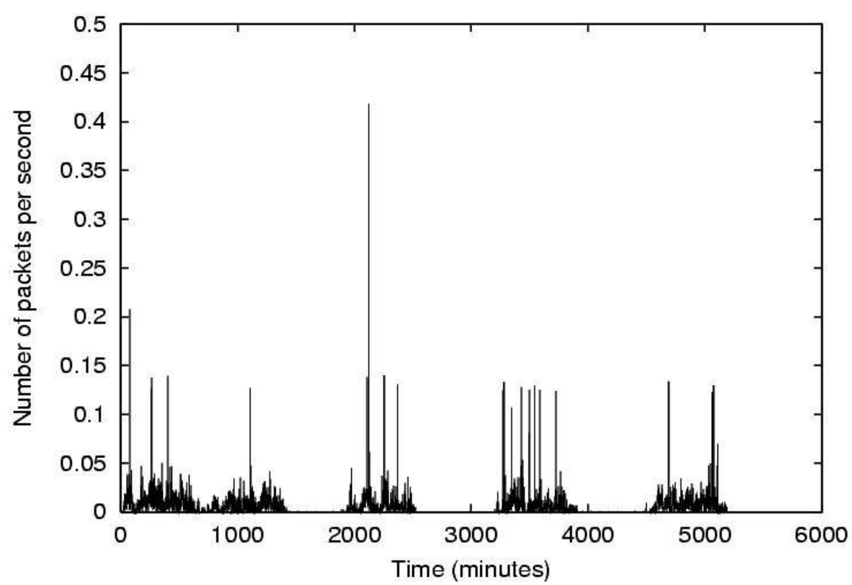
Attack Time Line



Training set example

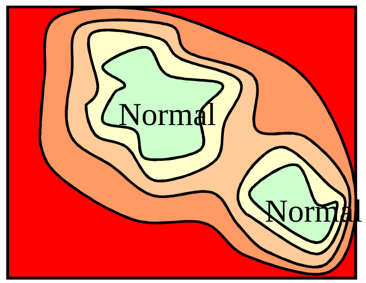


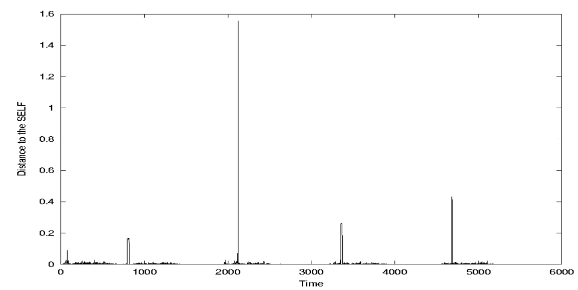
Test set example



Immune Anomaly Detection

Immune Approach





Active Sensors

Descriptor extractor

Experiments on real data:

•Network data from MIT-Darpa IDS evaluation.

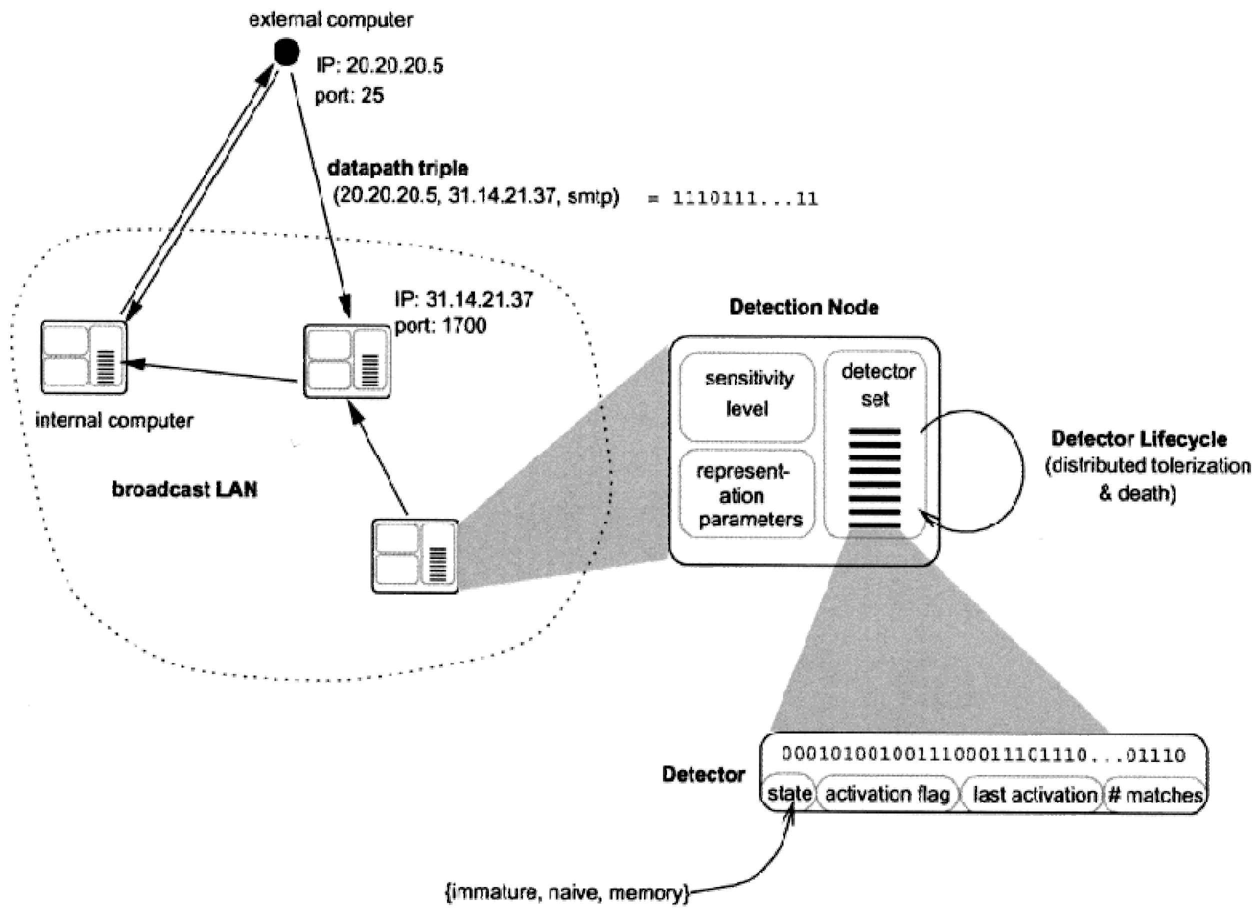
•Our system was able to detect network attacks using only positive data for training.

Peek Number	Attack Name	Attack Type
1	Back	DOS
2	Portsweep	PROBE
3	Satan	PROBE
4	Portsweep	PROBE
5	Neptune	DOS

Performance Evaluation

- **Tested using intrusion detection data sets from DARPA IDS evaluation program**
  - **CIDS used between normal (self) and abnormal (non-self) network events.**
    - **Evolved a set of fuzzy rules for attack detection.**
  - **Does not employ attack signatures**
  - **Highly scalable approach**
  - **System identified 87.5% of attacks with maximum 1% false alarm rate, 98.2% of attacks with a maximum 1.9% false alarm rate.**

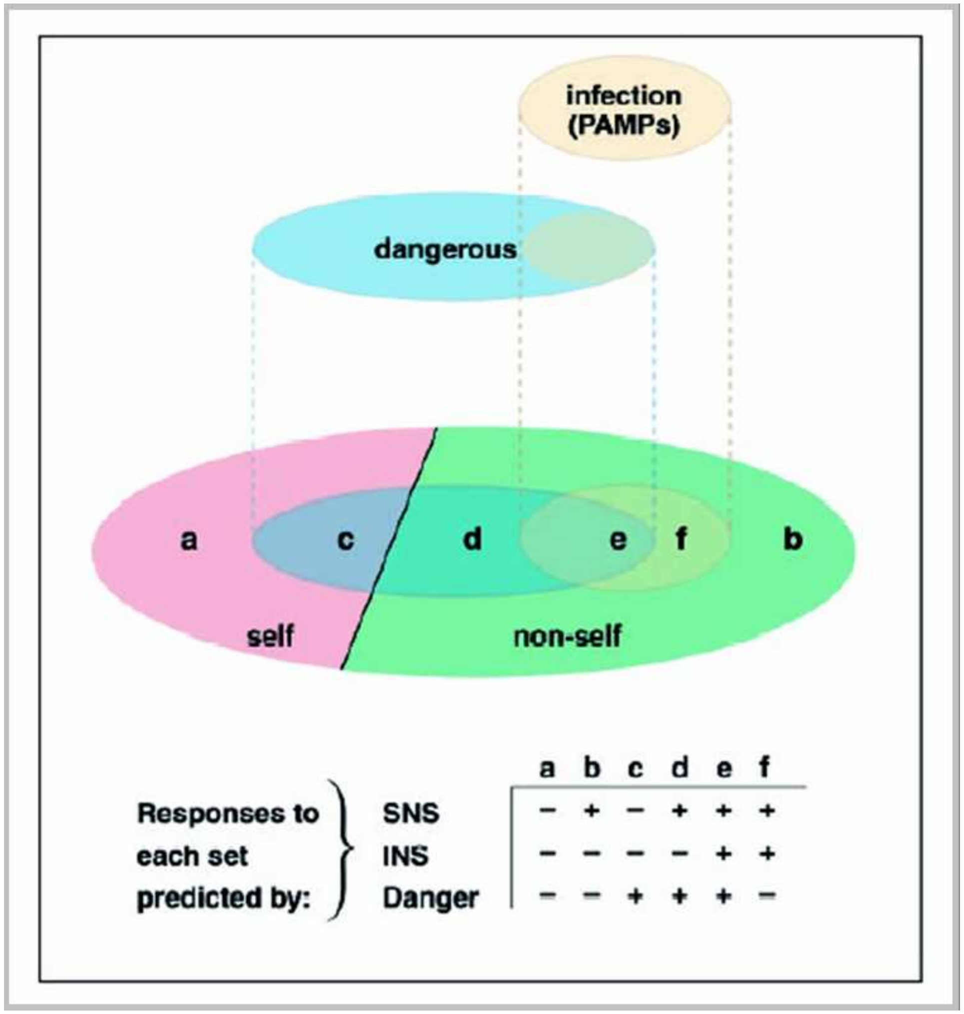
A Sense of Self: (Hofmeyr & Forrest 2000)



Advantages of Negative Selection

- From an information theory point of view, to characterize the normal space is equivalent to characterize the abnormal space.
- Distributed detection: Different set of detectors can be distributed at different location
- Other possibilities
  - Generalized and specialized detectors
  - Dynamic detector sets
  - Detectors with specific features
  - Artificial attack signatures

The Danger Model (Matzinger 1994,2002)



Partition of the Universe of Antigens

SNS:
self and nonself (*a and b*)

INS:
noninfectious self (*a*) and infectious nonself (*f*)

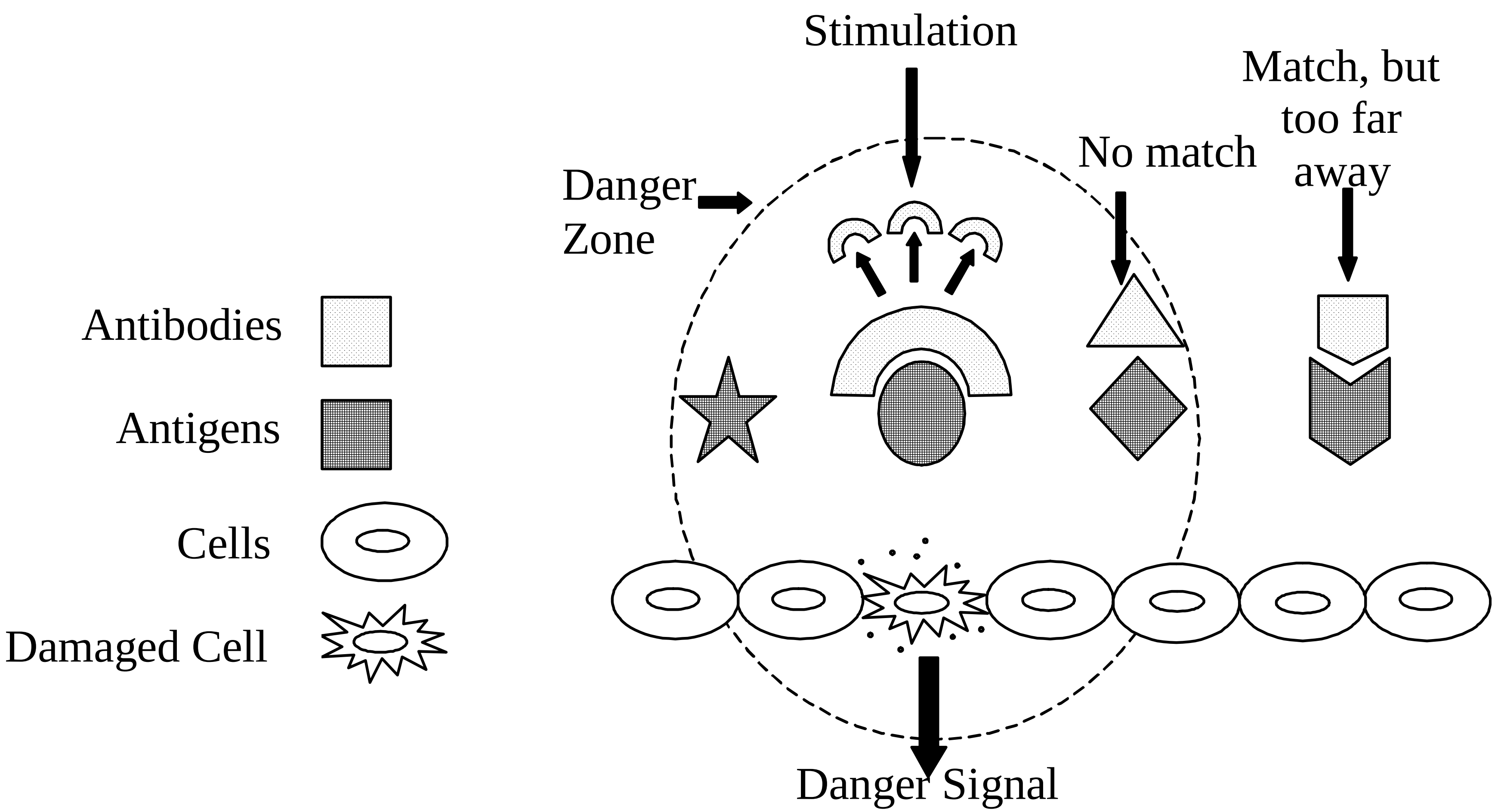
Danger model:
dangerous entities (*c, d, e*) and harmless ones

Danger Signal/ Harm indicator => Tissue Damage

Danger Theory in Intrusion Detection

- Need for discrimination: What should be responded to?
- Self-Nonself discrimination useful.
- Respond to Danger not to “foreignness”.
- Danger is measured by damage / distress signals.
- What would be ‘danger signals’?

The concept of Danger Zone (Uwe 2005)

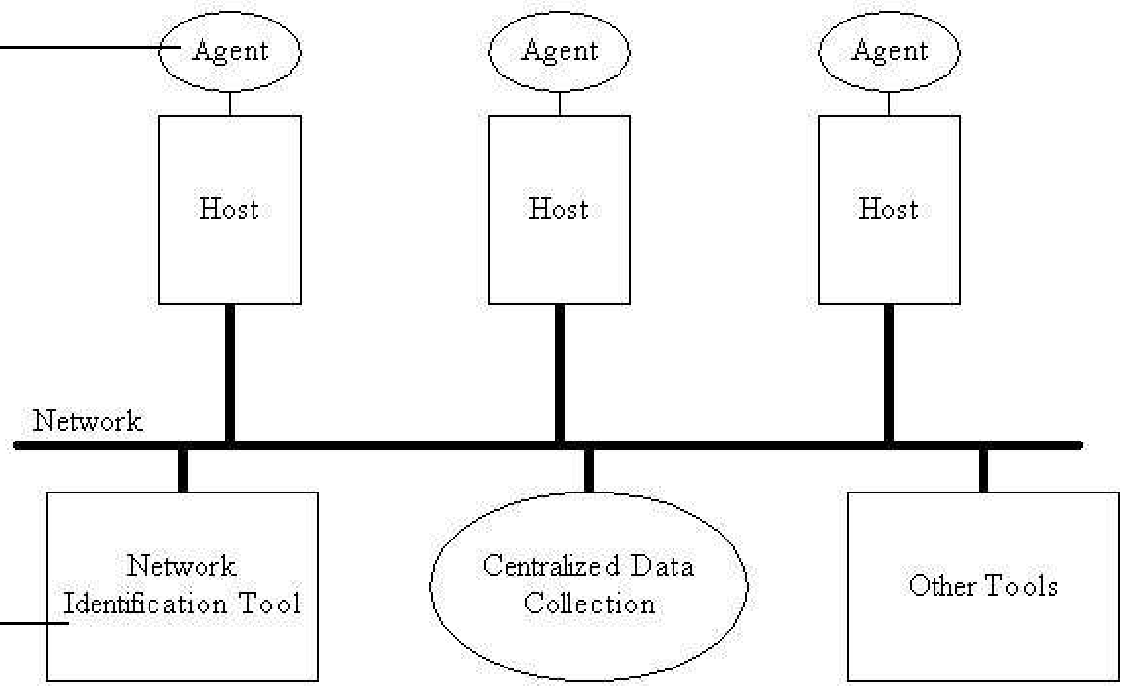


Agent Technology in Intrusion Detection

(Intelligent/Autonomous/Mobile)

Distributed Autonomous Agents

Agents monitor hosts & communicating with each other



recognizes coordinated attacks distributed through network

- Two main factors of *alert level = danger * transferability*
  - Danger (5 levels: minimal, cautionary, noticeable, serious, catastrophic)
  - Transferability (3 levels: none (local environment), partial, full)
- 3 alert levels: normal, partial alert, full alert
- Use neural networks with 8 features from statistics over time

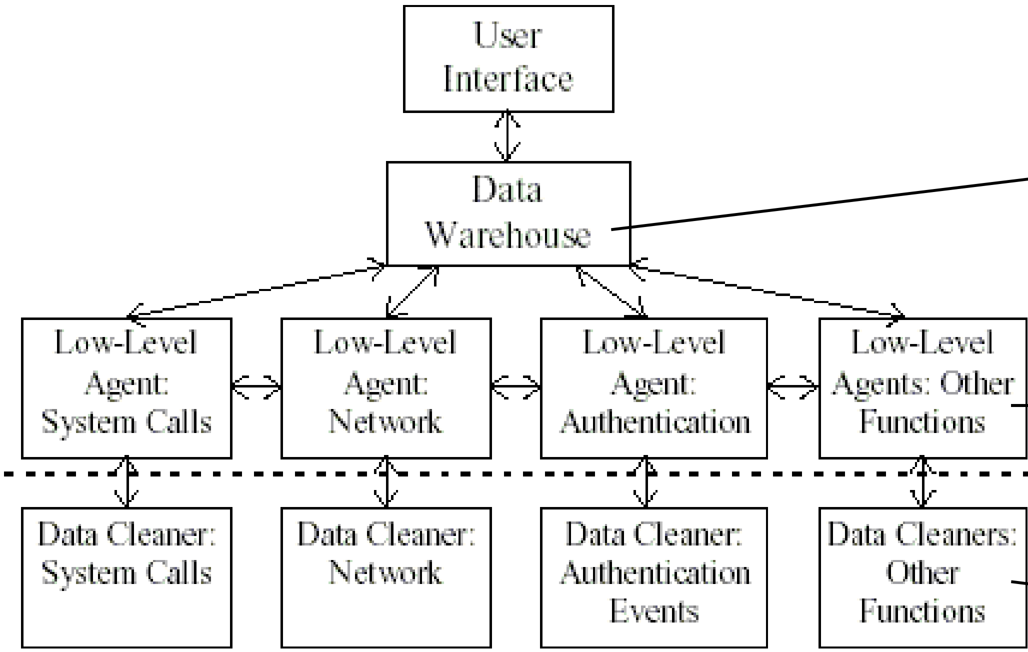
(J. Barrus, N. Rowe, A Distributed Autonomous-Agent NID and Response System,, 1998.)

AAFID - Autonomous Agents for ID

- AAFID components
  - *agents* monitor for interesting events, send messages to transceiver, may evolve over time using Genetic Programming (GP), may migrate from host to host
  - *filters* - data selection and data abstraction layer for agents that specify which records they need and what data format
  - *transceivers* – control (keeps track of agent execution) and data processing (process info from agents)
  - *monitors* – control and data processing from different hosts
- GP agents are trained on generated scenarios, where each agent is assigned a *fitness score* according to its accuracy

(E. Spafford, D. Zamboni, Intrusion Detection using Autonomous Agents, Computer Networks, 2001.)

Intelligent Agents for NID



IDS Architecture

Intelligent agents maintain DW by combining knowledge and data. Apply DM algorithms to discover global, temporal view

Lower-level agents – 1. ID level travel to cleaning agents, gather information, classify data

Distributed data cleaning agents

- System call traces data set
- RIPPER – classification algroithm

(G. Helmer, Intelligent Agents for Intrusion Detection, 1999.)

Immunity-Based IDS

- An Agent based approach for monitoring and detecting attacks
- A self adaptive system that can perform real-time detection of attacks
- It uses intelligent decision support modules for intrusion detection
- Provides a hierarchical security agent framework
- Each agent performs a unique function to address various security issues
- A Fuzzy decision support system is used to generate rules for attack detection

(Dasgupta 1998)

Role of Agents

- **Monitoring agents**: task for these agents are to look for malfunctions, anomalies, faults, intrusive activities in networked nodes
- Some agents work in the complement space (non-self) to monitor changes, others have special attack markers.

Other Type of Agents

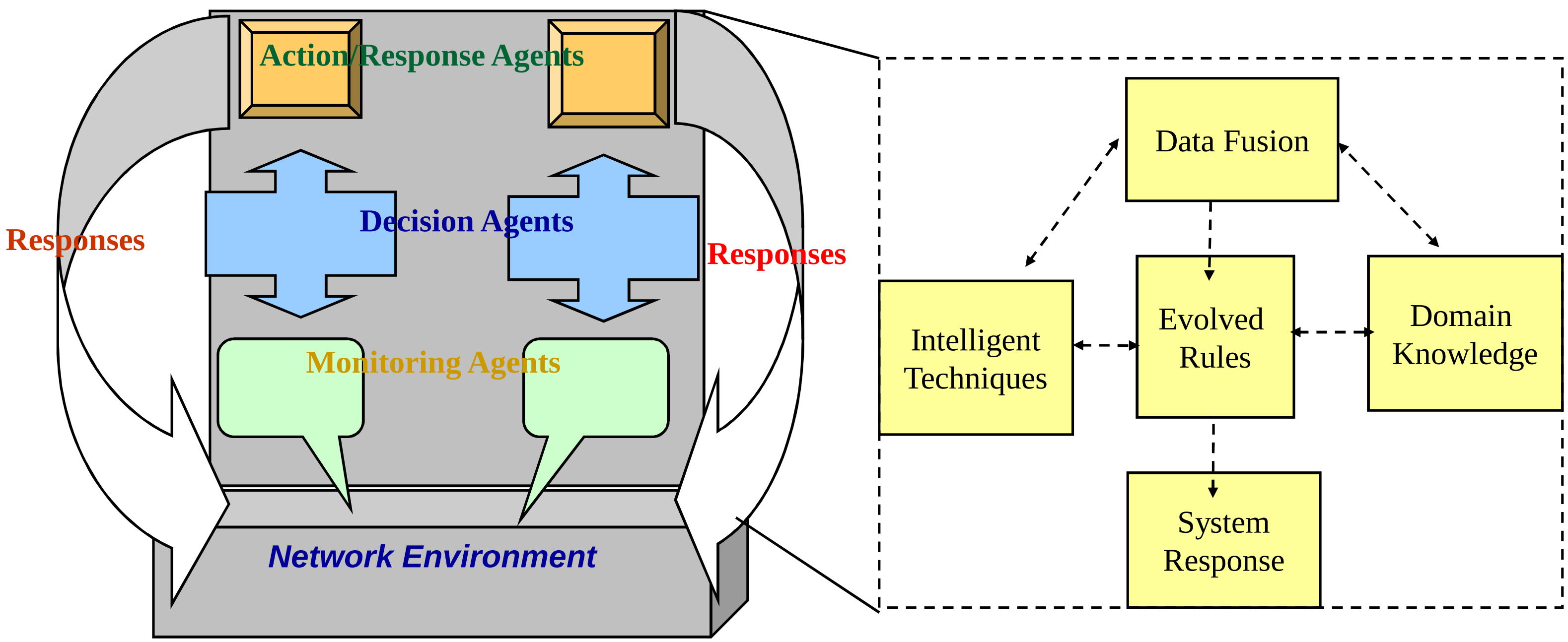
- **Communicator Agents**
  - Serve as message carriers or negotiators of other agents.
- **Decision Agents**
  - Involve in decision making using different intelligent techniques
- **Action Agents**

  activating specific agents according to the underlying security policies.
  - ***Helper Agents*:**
    - Reporting status of the intrusive activities to the end user
  - ***Killer Agents***
    - Takes drastic action in case of damaging malicious activities.
  - ***Suppressor Agents***
    - Suppress further actions taken by other agents in case false positives.

IIDS Agent Design

- Three main logical modules:
  - Monitor Agents
  - Decision Agents
  - Response Agents

Multi-Agent Architecture



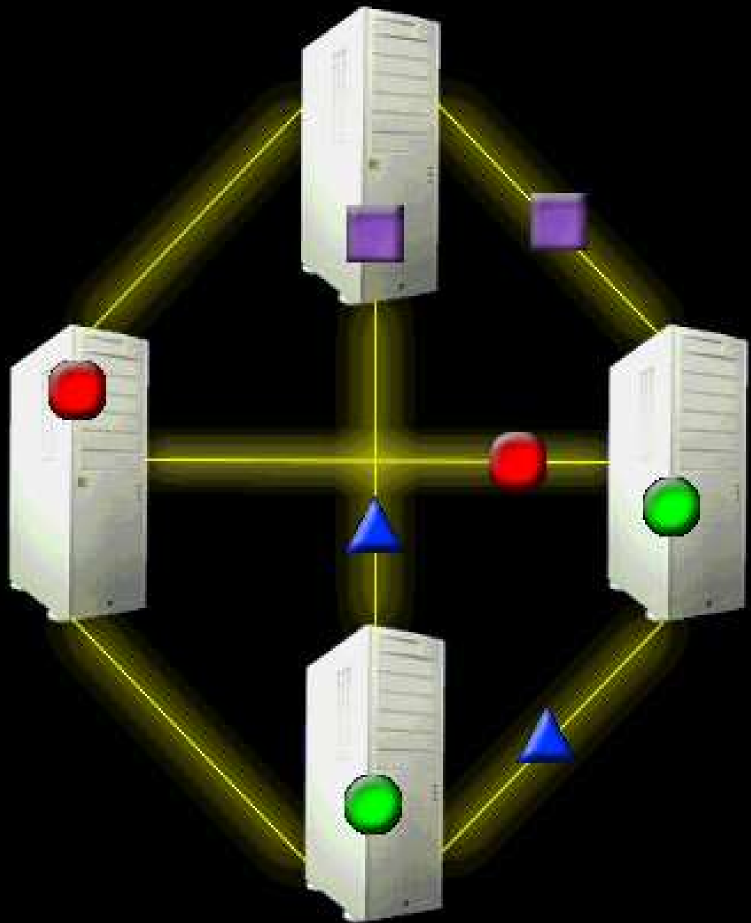
Components of decision Support Module

Role of Action/Response Agents

Depending on the nature of intrusive activities, these agents will take one of the following actions:
(based polices and preferences of the organization)

A1. Informing the system administrator via e-mail or other messaging system
A2. Change the priority of user processes
A3. Change access privileges of certain user
A4. Block a particular IP address or sender
A5. Disallow establishing a remote connection request
A6. Termination of existing network connection
A7. Restarting of a particular machine
A8. Logout user or close session

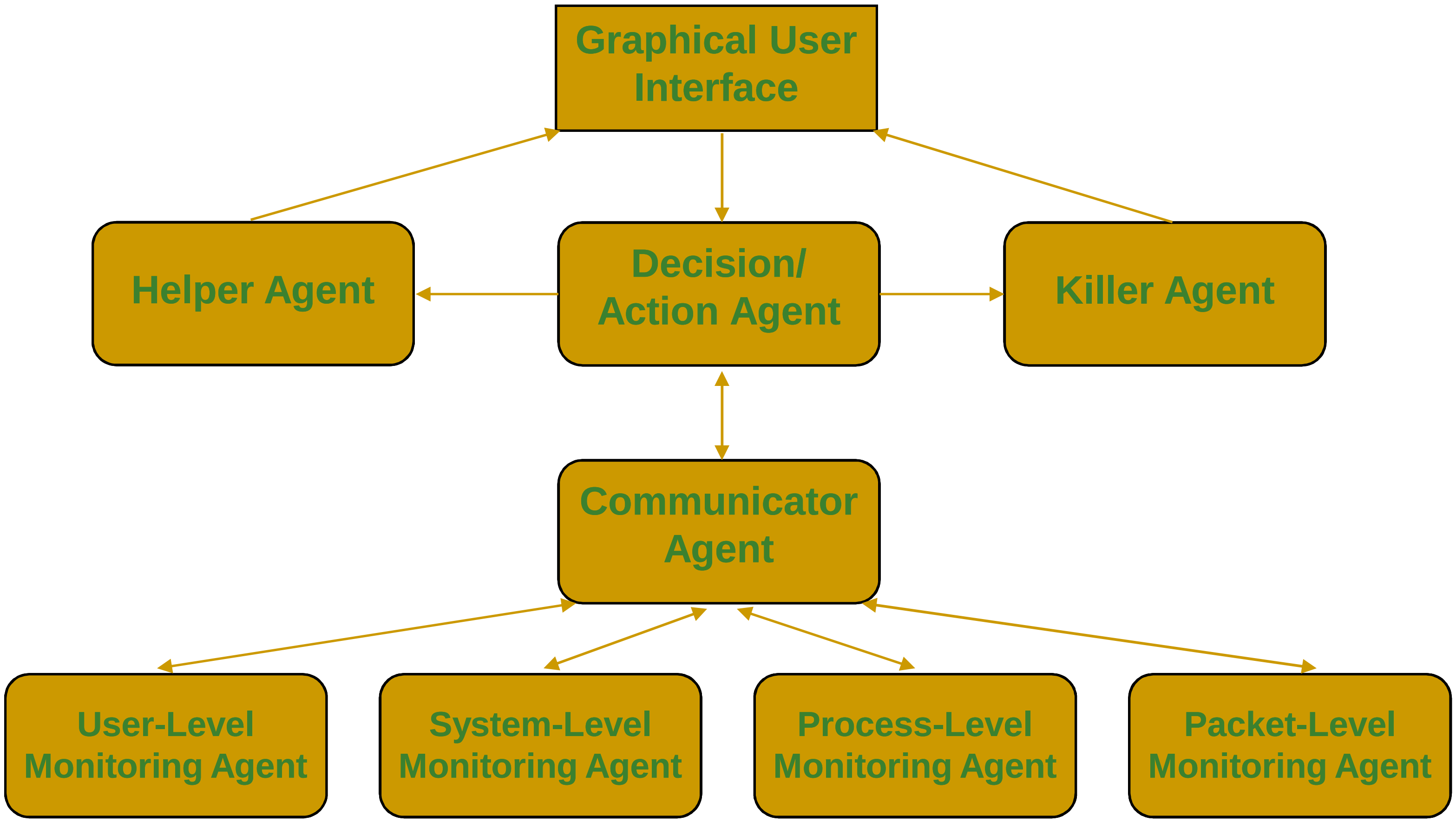
Mobile Security Agents

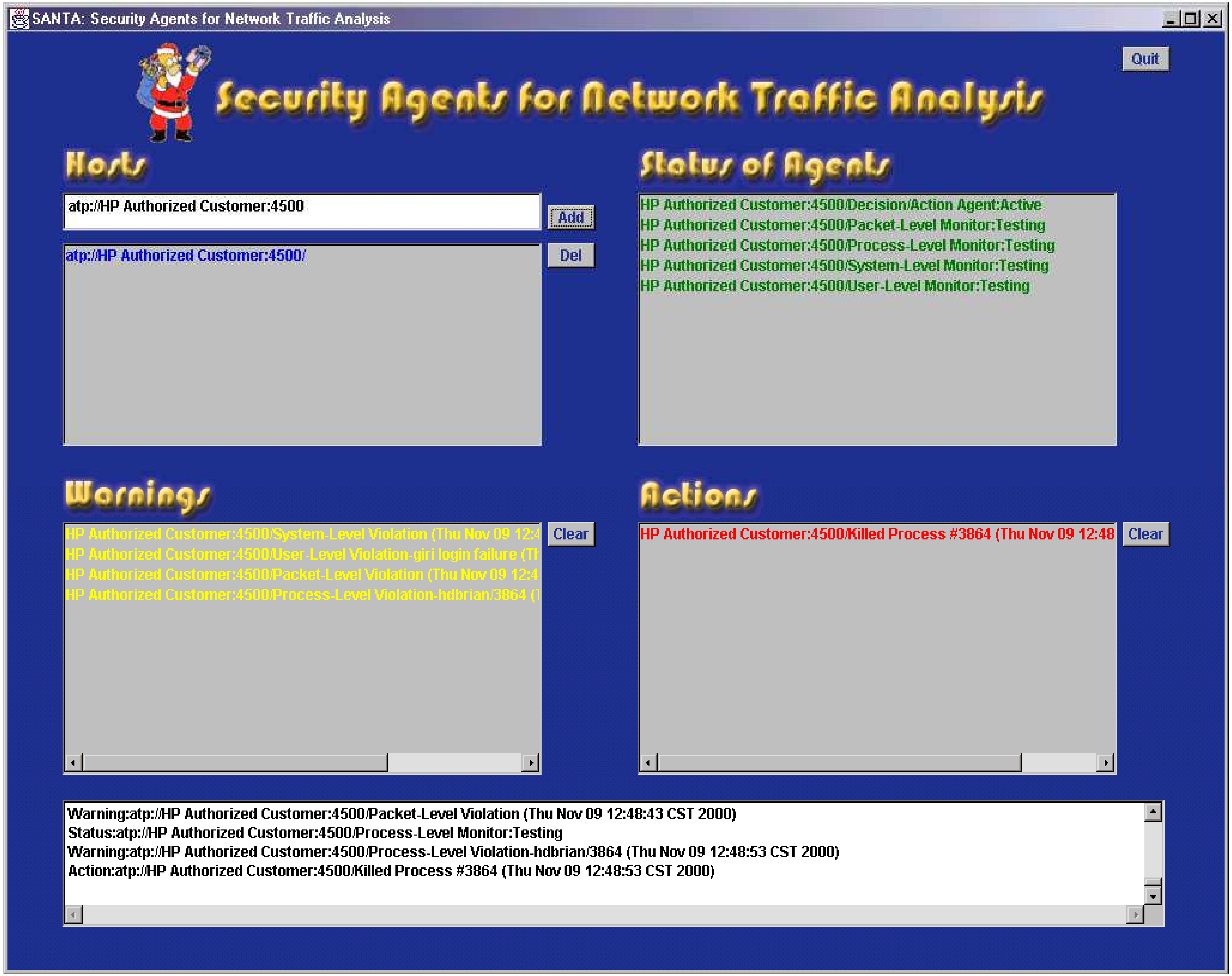


"Immunity-based agents roam around the machines (nodes or routers) and monitor the situation in the network (i.e., look for changes such as malfunctions, faults, abnormalities, misuse, deviations, intrusions, etc.). These agents can mutually recognize each other's activities and can take appropriate actions according to the underlying security policies. Such an agent can learn and adapt to its environment dynamically and can detect both known and unknown intrusions."

- Dr. Dipankar Dasgupta

SANTA: Mobile Agent Architecture



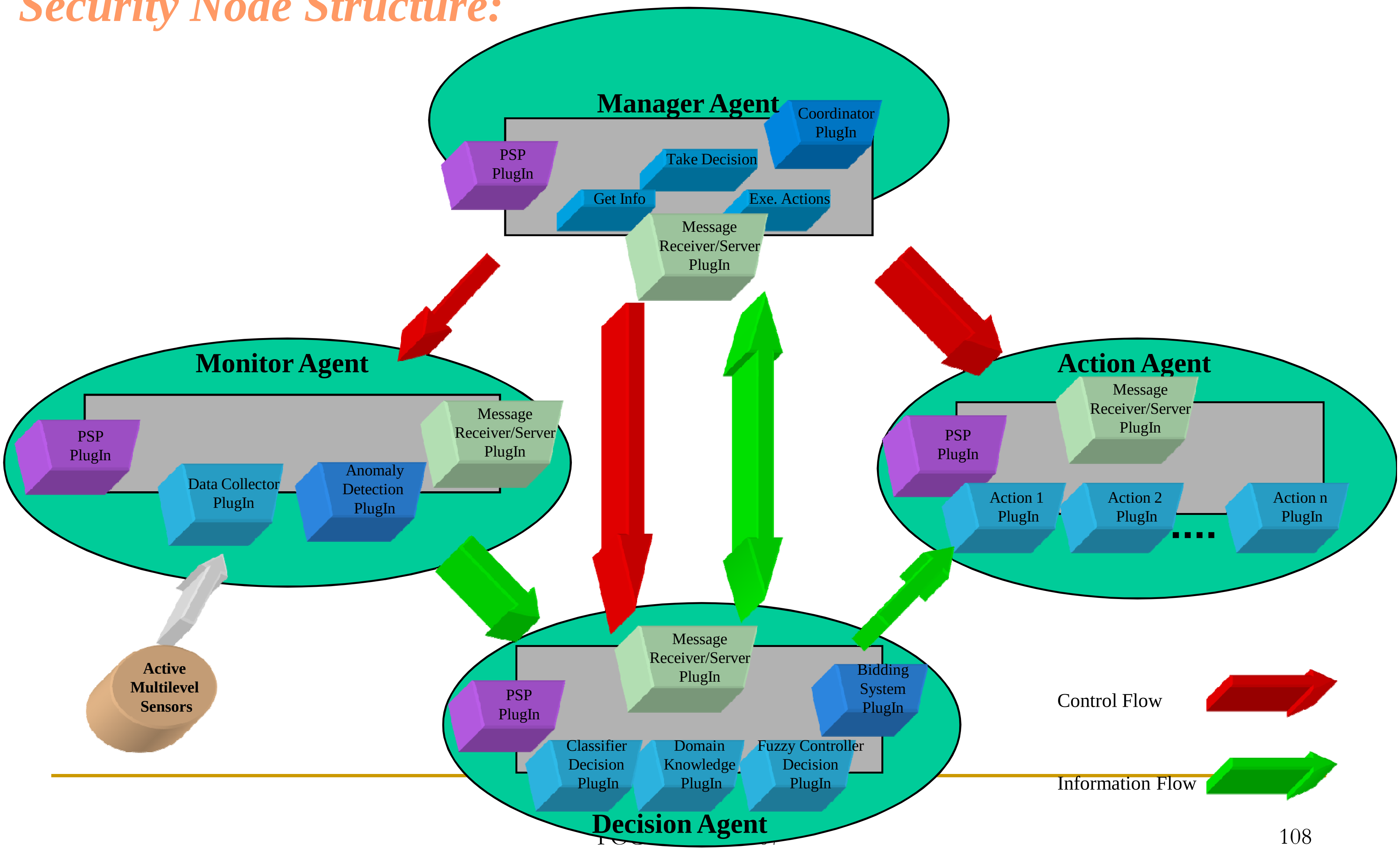

SANTA: Security Agents for Network Traffic Analysis
Quit
Security Agents for Network Traffic Analysis
Hosts
atp://HP Authorized Customer:4500
Add
atp://HP Authorized Customer:4500/
Del
Status of Agents
HP Authorized Customer:4500/Decision/Action Agent:Active
HP Authorized Customer:4500/Packet-Level Monitor:Testing
HP Authorized Customer:4500/Process-Level Monitor:Testing
HP Authorized Customer:4500/System-Level Monitor:Testing
HP Authorized Customer:4500/User-Level Monitor:Testing
Warnings
HP Authorized Customer:4500/System-Level Violation (Thu Nov 09 12:4
HP Authorized Customer:4500/User-Level Violation-giri login failure (Th
HP Authorized Customer:4500/Packet-Level Violation (Thu Nov 09 12:4
HP Authorized Customer:4500/Process-Level Violation-hdbrian/3864 (
Clear
Actions
HP Authorized Customer:4500/Killed Process #3864 (Thu Nov 09 12:48
Clear
Warning:atp://HP Authorized Customer:4500/Packet-Level Violation (Thu Nov 09 12:48:43 CST 2000)
Status:atp://HP Authorized Customer:4500/Process-Level Monitor:Testing
Warning:atp://HP Authorized Customer:4500/Process-Level Violation-hdbrian/3864 (Thu Nov 09 12:48:53 CST 2000)
Action:atp://HP Authorized Customer:4500/Killed Process #3864 (Thu Nov 09 12:48:53 CST 2000)

CIDS: Cougaar-Based Security Agents

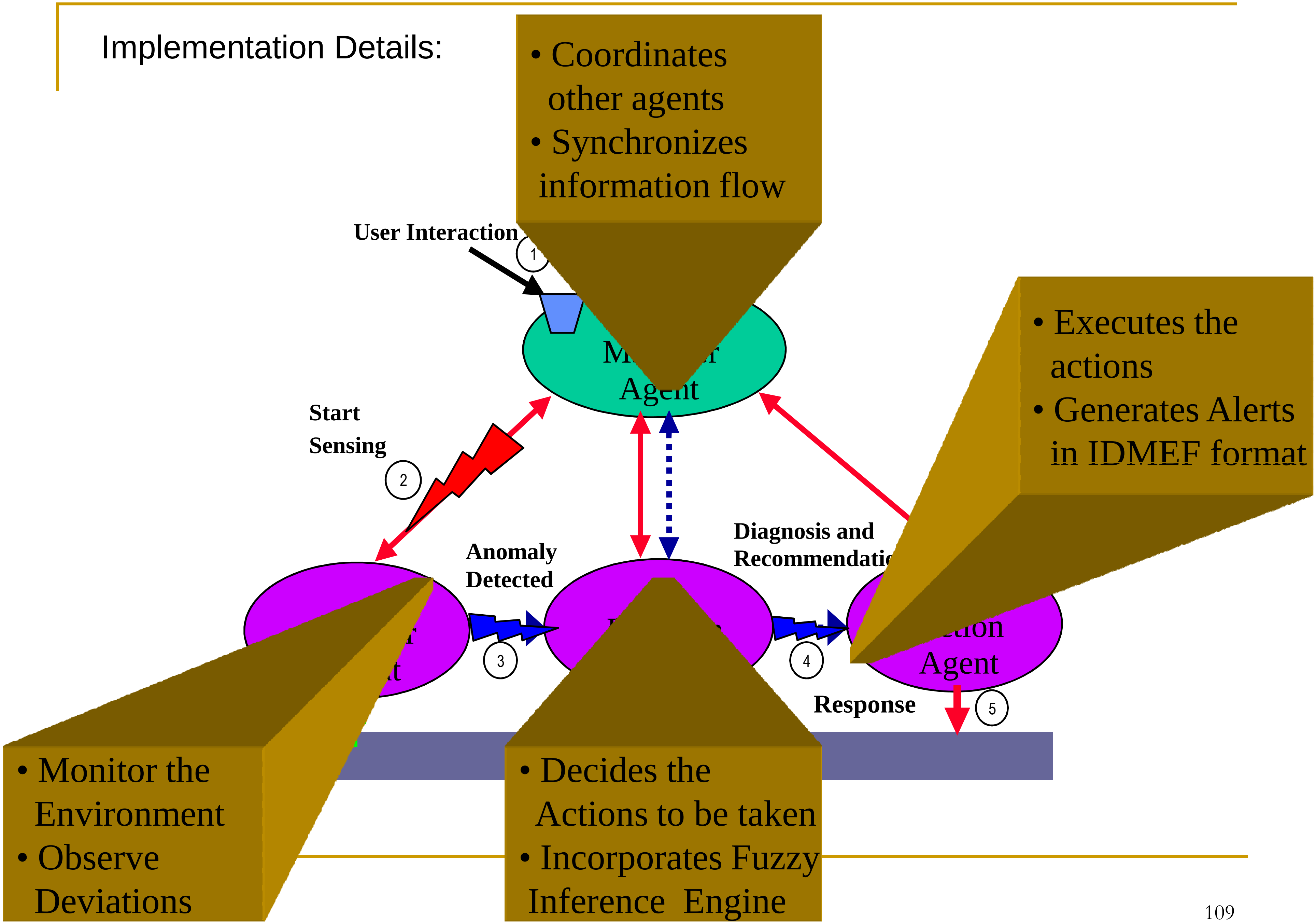
- Developing a multi-agent intrusion/ anomaly **detection** and **response** system.
- Monitor networked computer's activities at **multiple levels** (from packet to user-level).
- Agents are Autonomous having properties like **Mobility, Adaptivity** and **Collaboration**.
- Agents are highly distributed, but activities are coordinated in an hierarchical fashion.

CIDS: A Security Agent Architecture

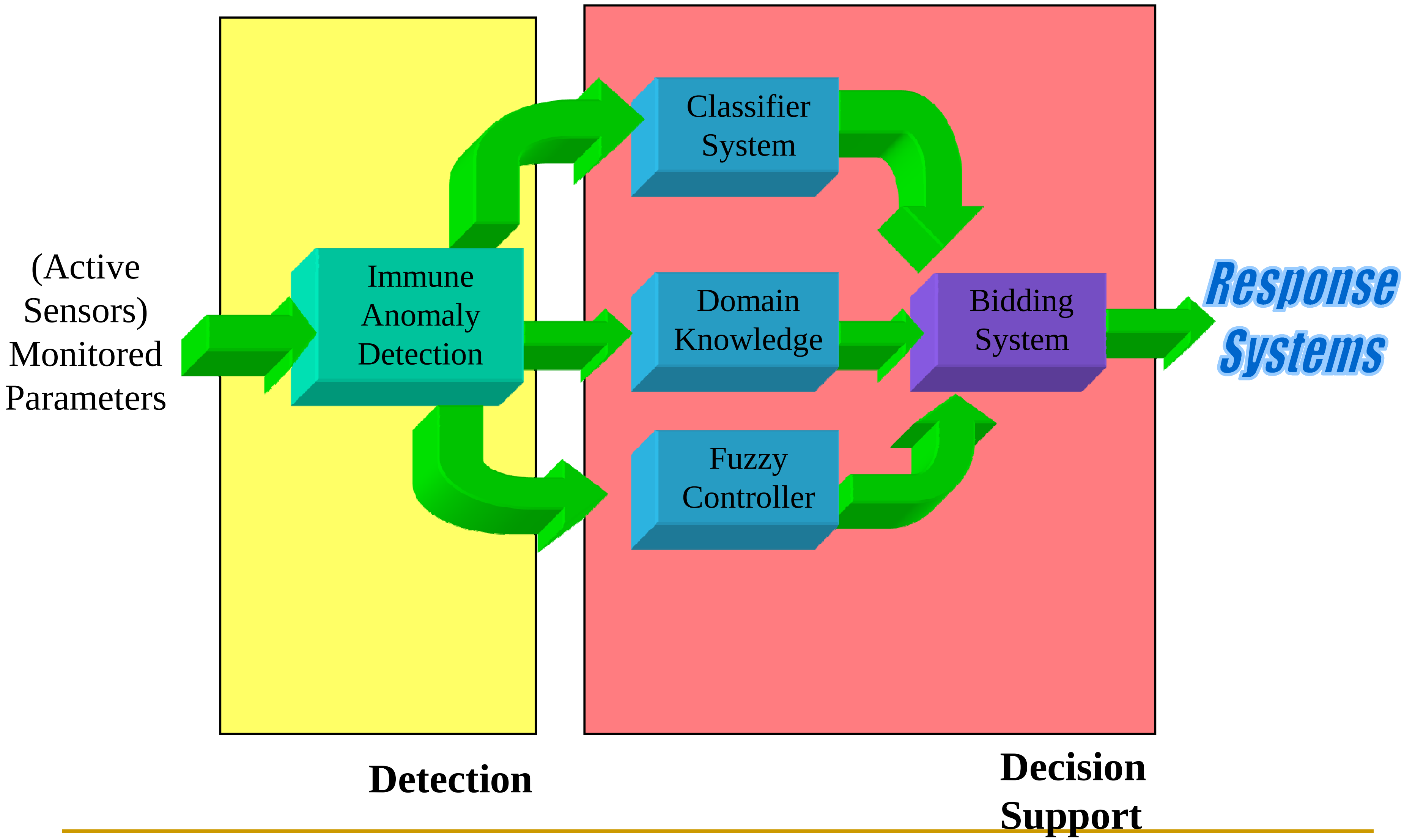
Security Node Structure:



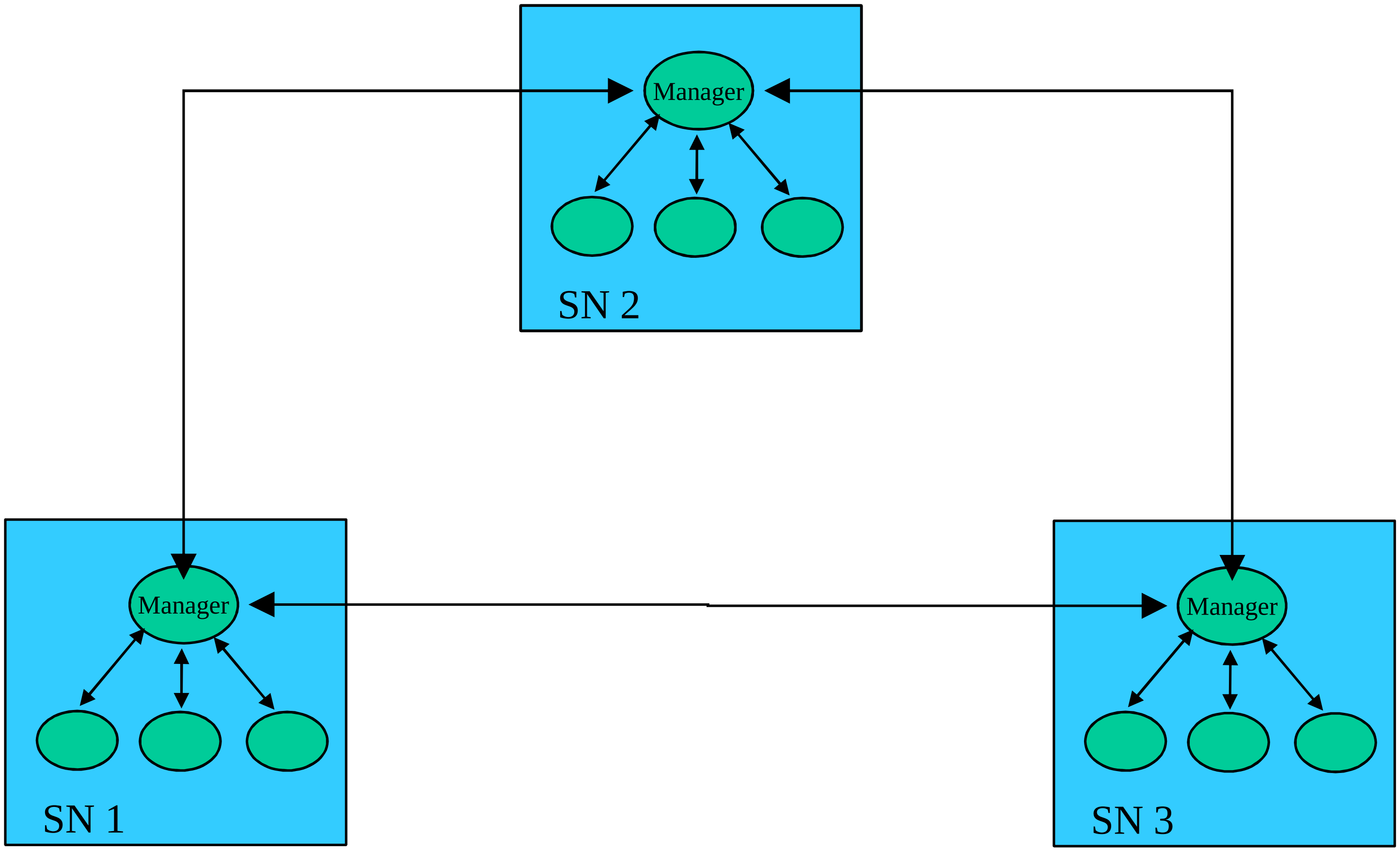
Implementation Details:



Intelligent Decision Processes

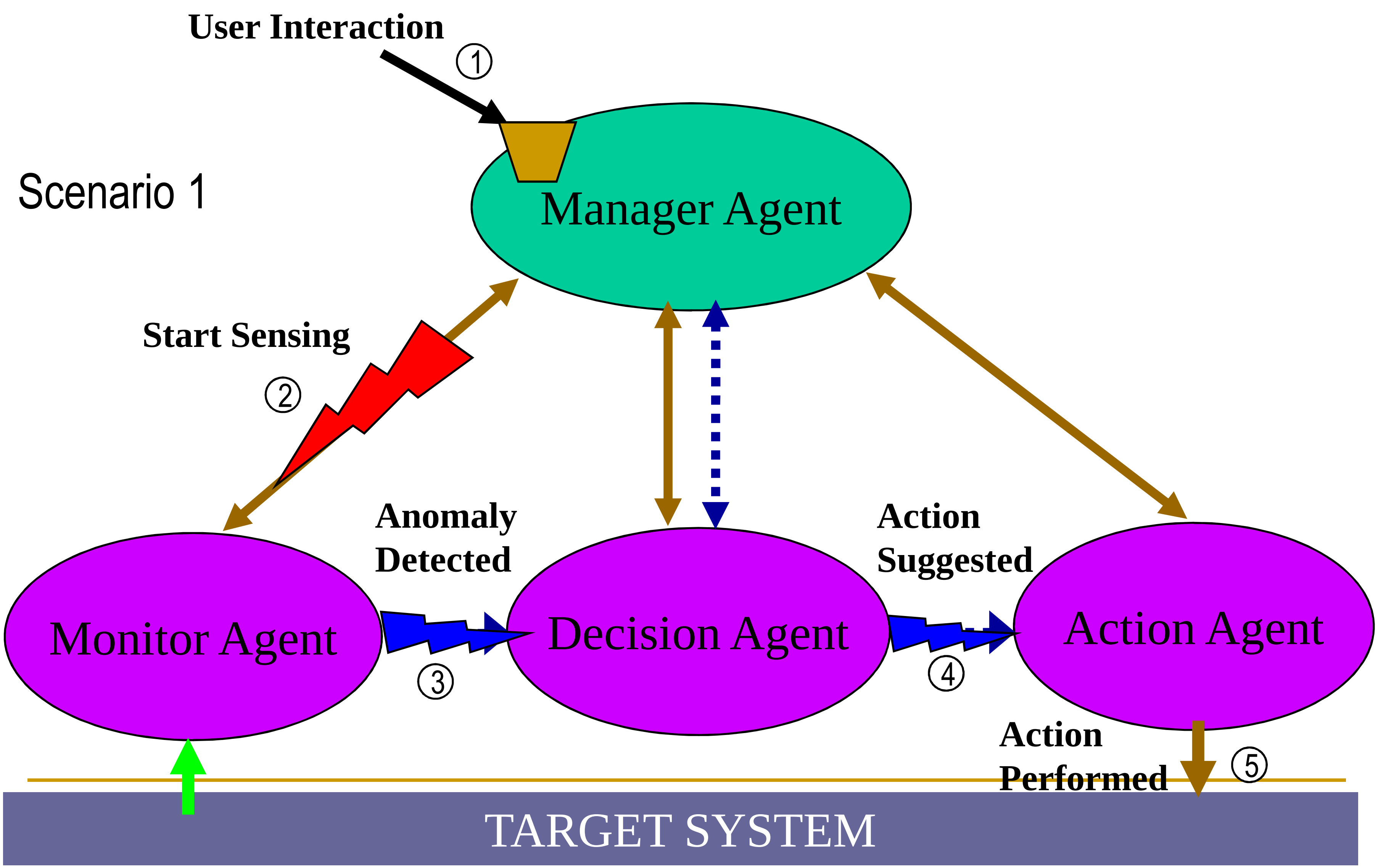


Security Agents Society

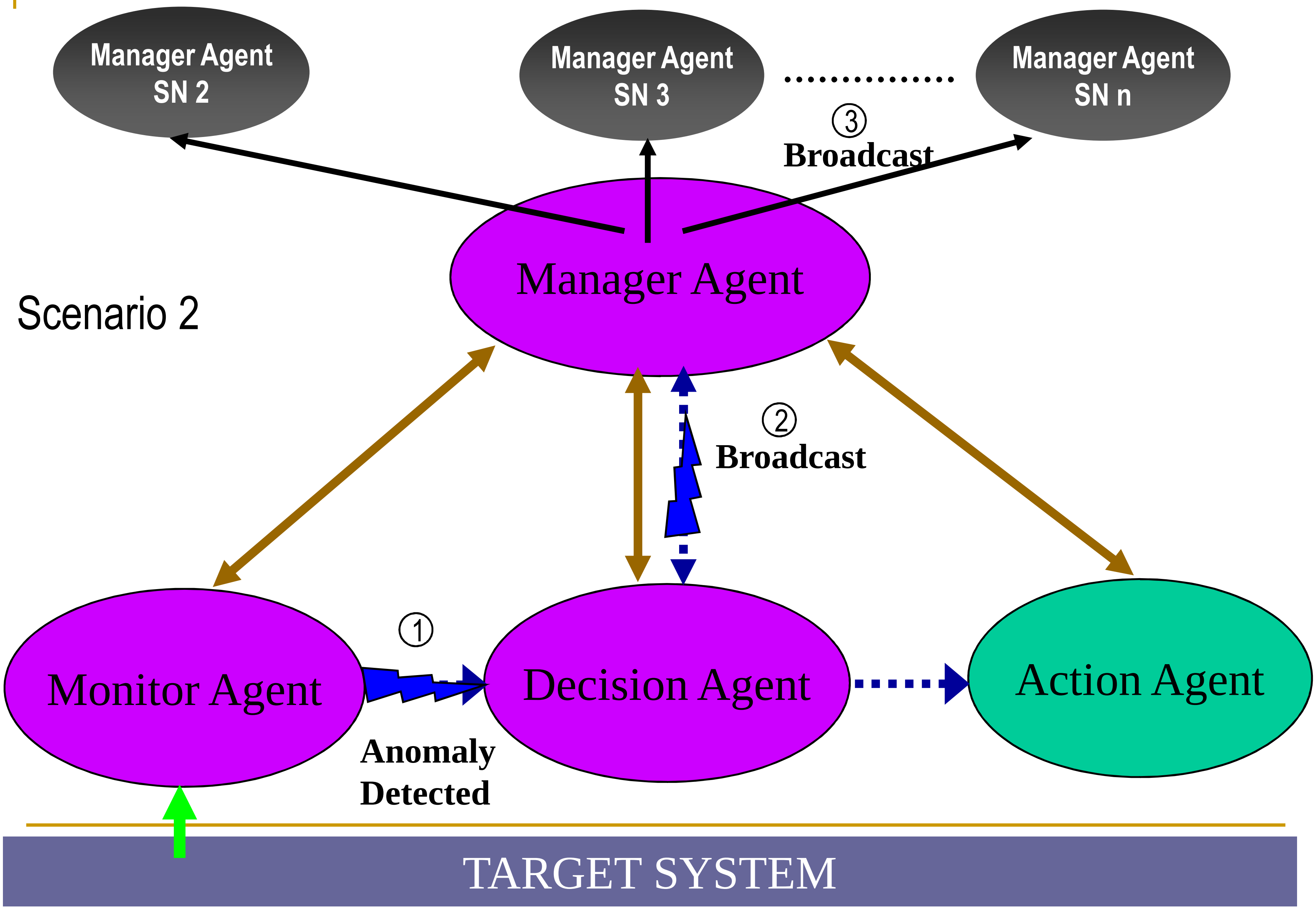


Sequence of Operations

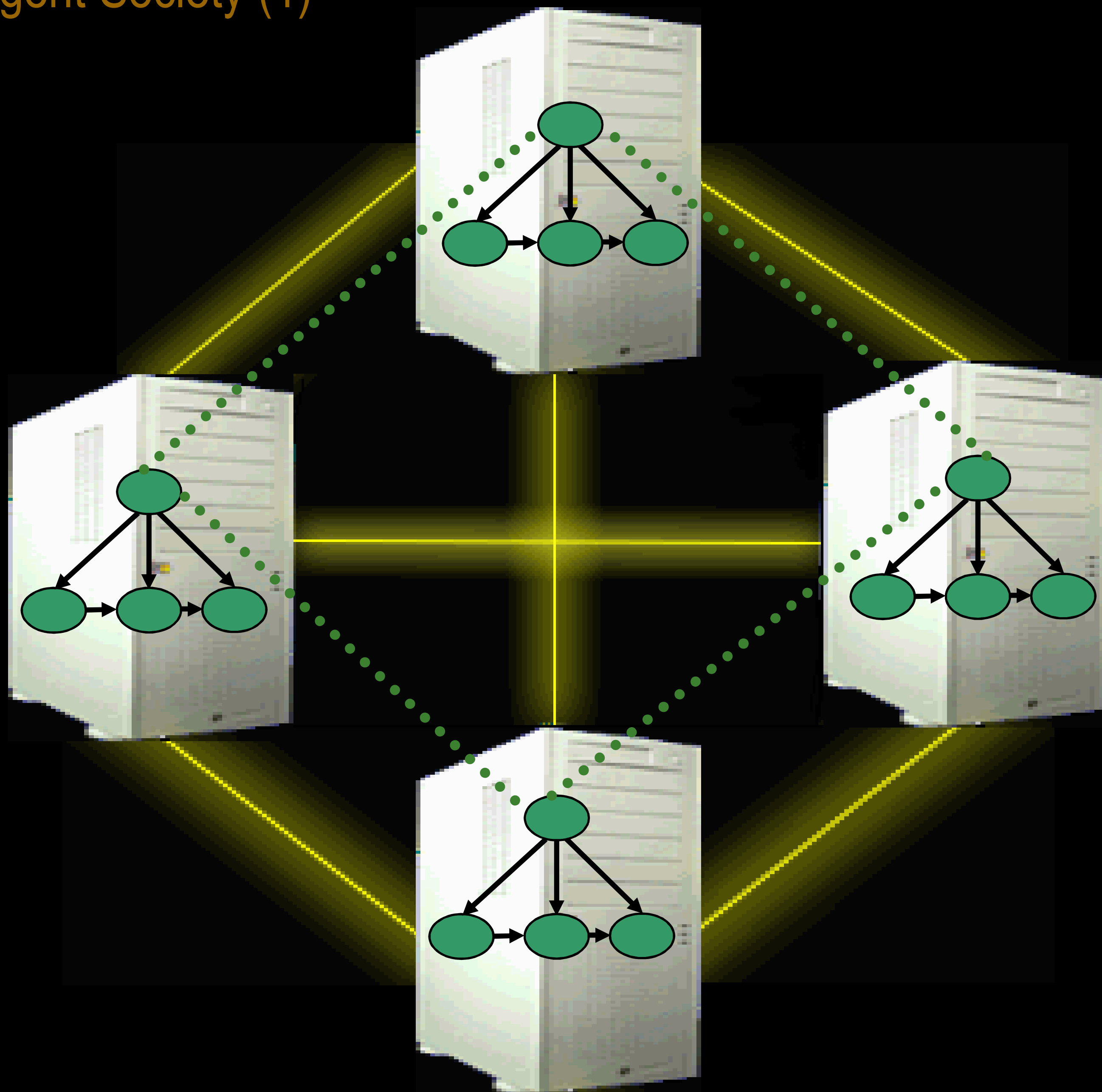
Scenario 1



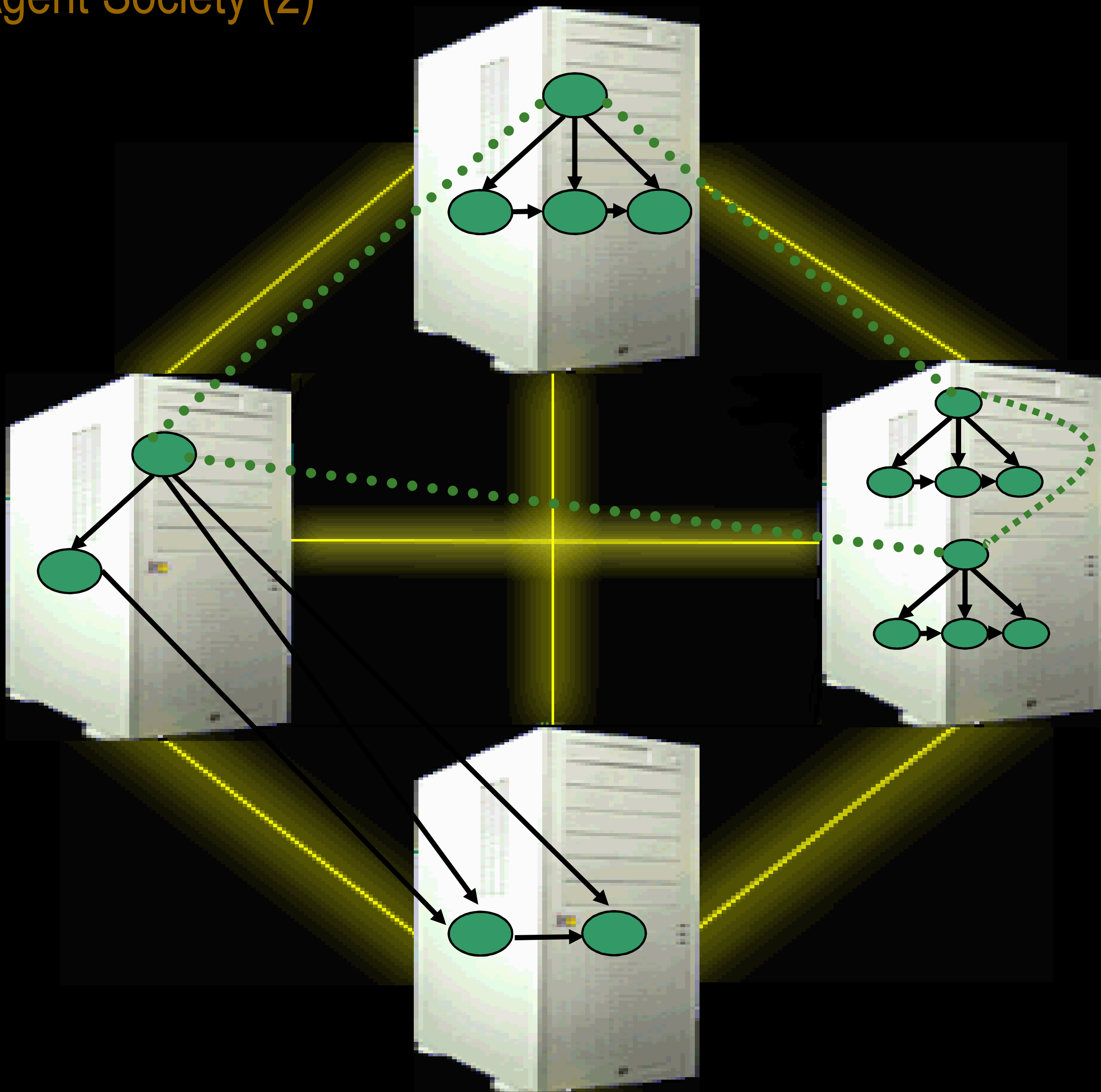
Sequence of Operations

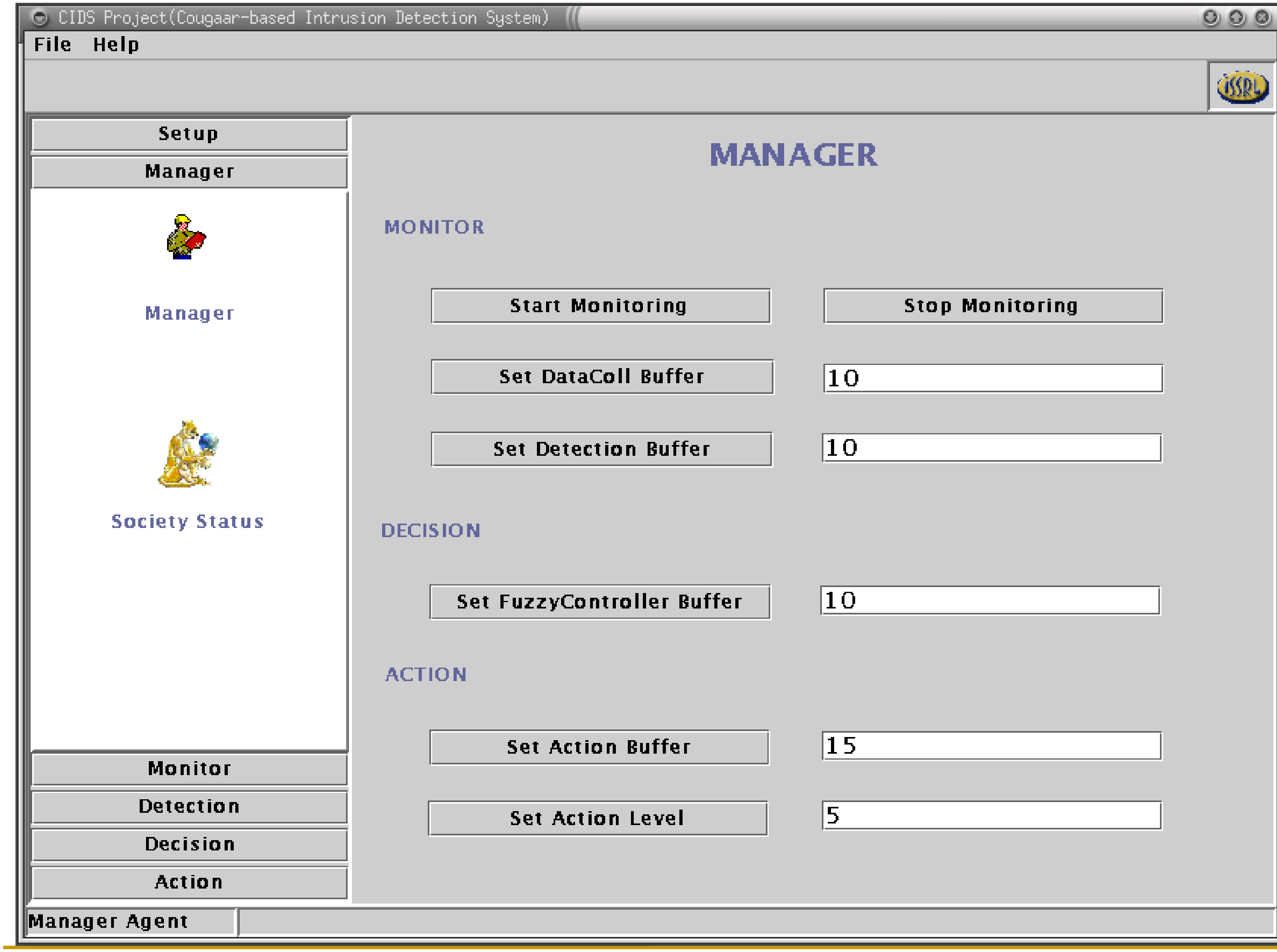


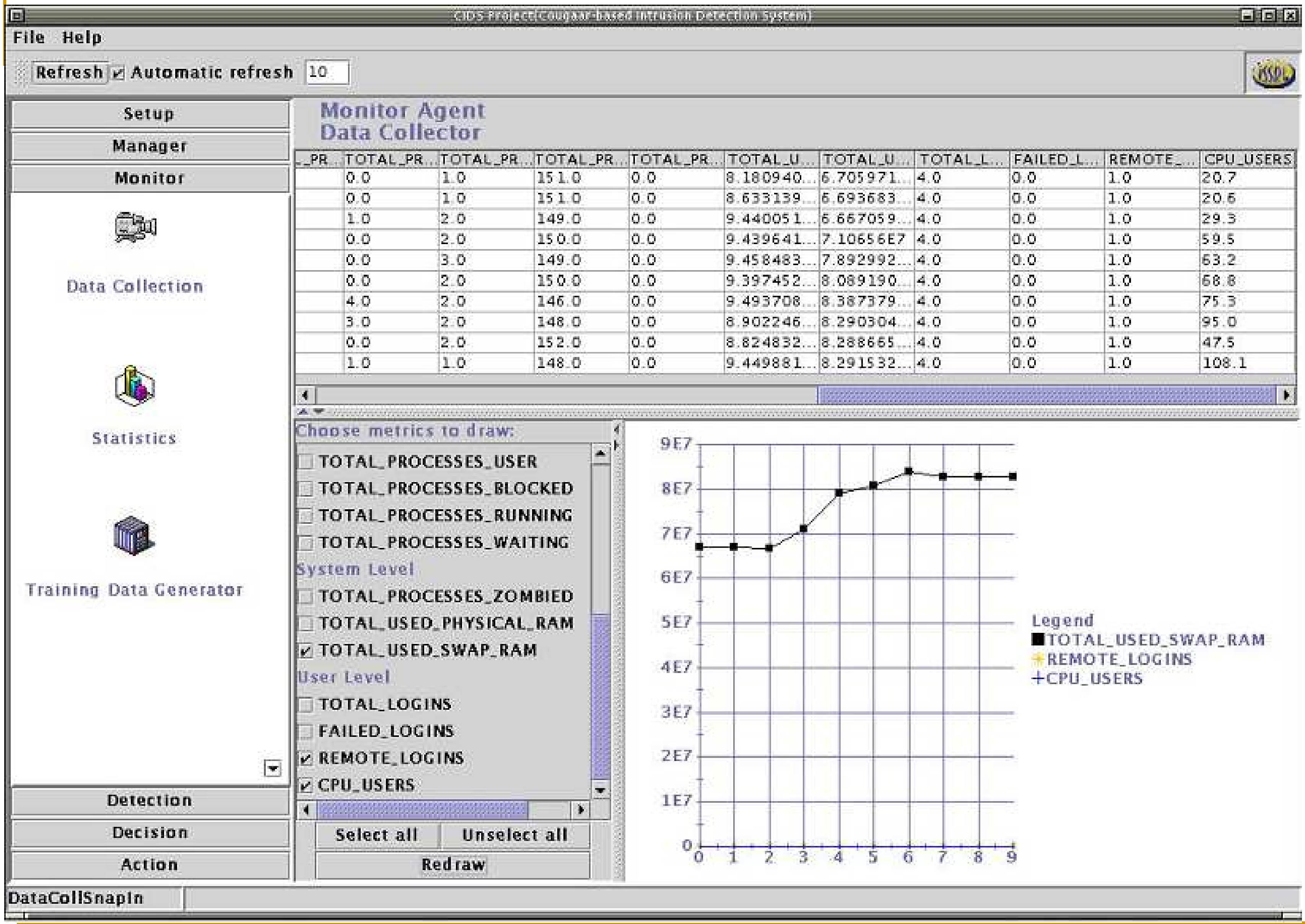
Security Agent Society (1)

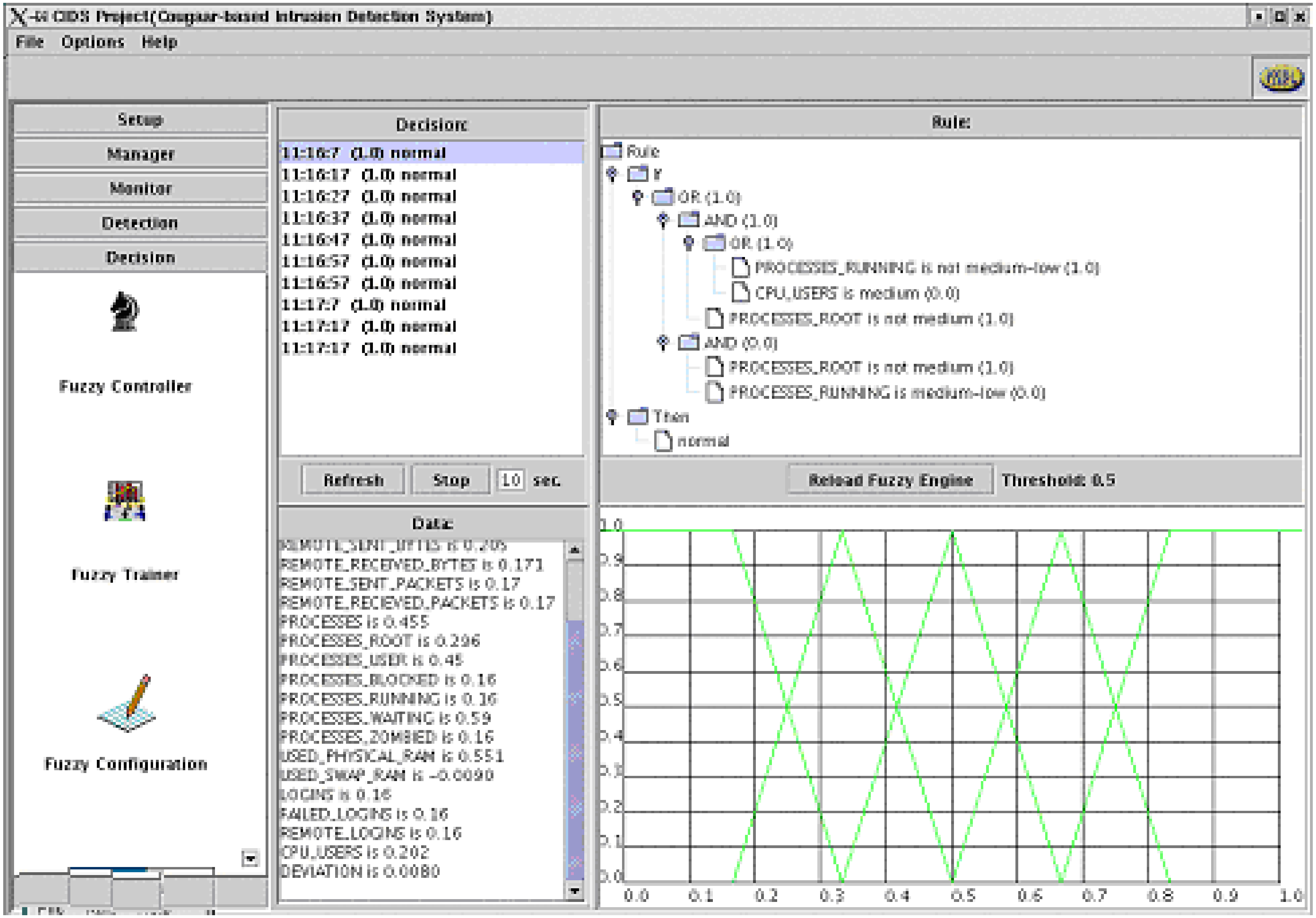


Security Agent Society (2)



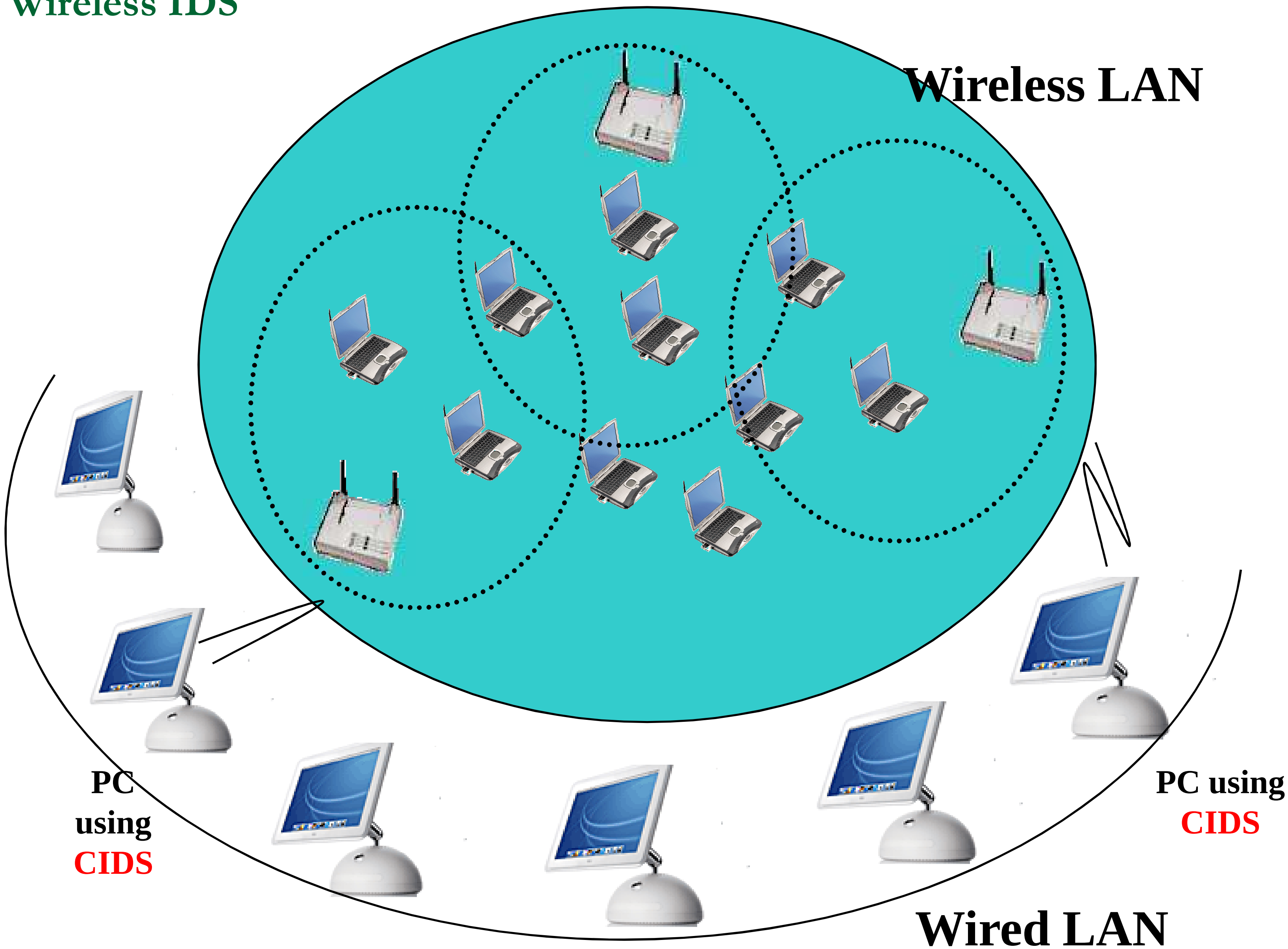

CIDS Project(Cougaar-based Intrusion Detection System)
File
Help
Setup
Manager
Manager
Society Status
Monitor
Detection
Decision
Action
Manager Agent
MANAGER
MONITOR
Start Monitoring
Stop Monitoring
Set DataColl Buffer
10
Set Detection Buffer
10
DECISION
Set FuzzyController Buffer
10
ACTION
Set Action Buffer
15
Set Action Level
5


CIDS Project(Cougaar-based Intrusion Detection System)
File Help
Refresh
Automatic refresh
10
Setup
Manager
Monitor
Data Collection
Statistics
Training Data Generator
Detection
Decision
Action
Monitor Agent
Data Collector
_PR.. TOTAL_PR... TOTAL_PR... TOTAL_PR... TOTAL_PR... TOTAL_U... TOTAL_U... TOTAL_L... FAILED_L... REMOTE_... CPU_USERS
0.0 1.0 151.0 0.0 8.180940... 6.705971... 4.0 0.0 1.0 20.7
0.0 1.0 151.0 0.0 8.633139... 6.693683... 4.0 0.0 1.0 20.6
1.0 2.0 149.0 0.0 9.440051... 6.667059... 4.0 0.0 1.0 29.3
0.0 2.0 150.0 0.0 9.439641... 7.10656E7 4.0 0.0 1.0 59.5
0.0 3.0 149.0 0.0 9.458483... 7.892992... 4.0 0.0 1.0 63.2
0.0 2.0 150.0 0.0 9.397452... 8.089190... 4.0 0.0 1.0 68.8
4.0 2.0 146.0 0.0 9.493708... 8.387379... 4.0 0.0 1.0 75.3
3.0 2.0 148.0 0.0 8.902246... 8.290304... 4.0 0.0 1.0 95.0
0.0 2.0 152.0 0.0 8.824832... 8.288665... 4.0 0.0 1.0 47.5
1.0 1.0 148.0 0.0 9.449881... 8.291532... 4.0 0.0 1.0 108.1
Choose metrics to draw:
TOTAL_PROCESSES_USER
TOTAL_PROCESSES_BLOCKED
TOTAL_PROCESSES_RUNNING
TOTAL_PROCESSES_WAITING
System Level
TOTAL_PROCESSES_ZOMBIED
TOTAL_USED_PHYSICAL_RAM
TOTAL_USED_SWAP_RAM
User Level
TOTAL_LOGINS
FAILED_LOGINS
REMOTE_LOGINS
CPU_USERS
Select all
Unselect all
Redraw
9E7
8E7
7E7
6E7
5E7
4E7
3E7
2E7
1E7
0
0 1 2 3 4 5 6 7 8 9
Legend
TOTAL_USED_SWAP_RAM
REMOTE_LOGINS
CPU_USERS
DataCollSnapIn



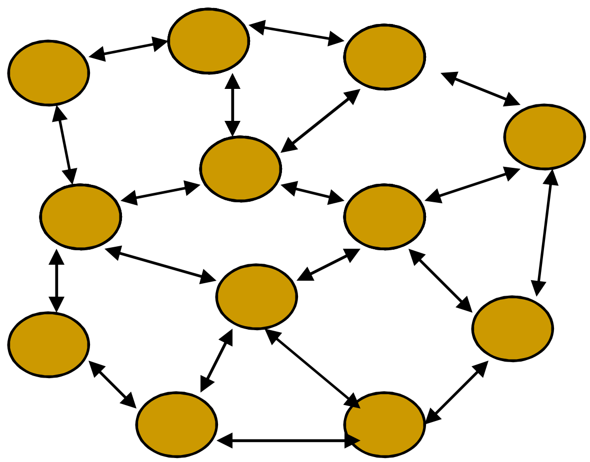
-ii CIDS Project(Cougaar-based Intrusion Detection System)
File Options Help
Setup
Manager
Monitor
Detection
Decision
Fuzzy Controller
Fuzzy Trainer
Fuzzy Configuration
Decision
11:16:7 (1.0) normal
11:16:17 (1.0) normal
11:16:27 (1.0) normal
11:16:37 (1.0) normal
11:16:47 (1.0) normal
11:16:57 (1.0) normal
11:16:57 (1.0) normal
11:17:7 (1.0) normal
11:17:17 (1.0) normal
11:17:17 (1.0) normal
Refresh
Stop
10 sec.
Data
REMOTE_RECEIVED_BYTES is 0.171
REMOTE_SENT_PACKETS is 0.17
REMOTE_RECIEVED_PACKETS is 0.17
PROCESSES is 0.455
PROCESSES_ROOT is 0.296
PROCESSES_USER is 0.45
PROCESSES_BLOCKED is 0.16
PROCESSES_RUNNING is 0.16
PROCESSES_WAITING is 0.59
PROCESSES_ZOMBIED is 0.16
USED_PHYSICAL_RAM is 0.551
USED_SWAP_RAM is -0.0090
LOGINS is 0.16
FAILED_LOGINS is 0.16
REMOTE_LOGINS is 0.16
CPU_USERS is 0.202
DEVIATION is 0.0080
Rule
Rule
If
OR (1.0)
AND (1.0)
OR (1.0)
PROCESSES_RUNNING is not medium-low (1.0)
CPU_USERS is medium (0.0)
PROCESSES_ROOT is not medium (1.0)
AND (0.0)
PROCESSES_ROOT is not medium (1.0)
PROCESSES_RUNNING is medium-low (0.0)
Then
normal
Reload Fuzzy Engine
Threshold: 0.5
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0

Wireless IDS



Large Scale Network Survivability

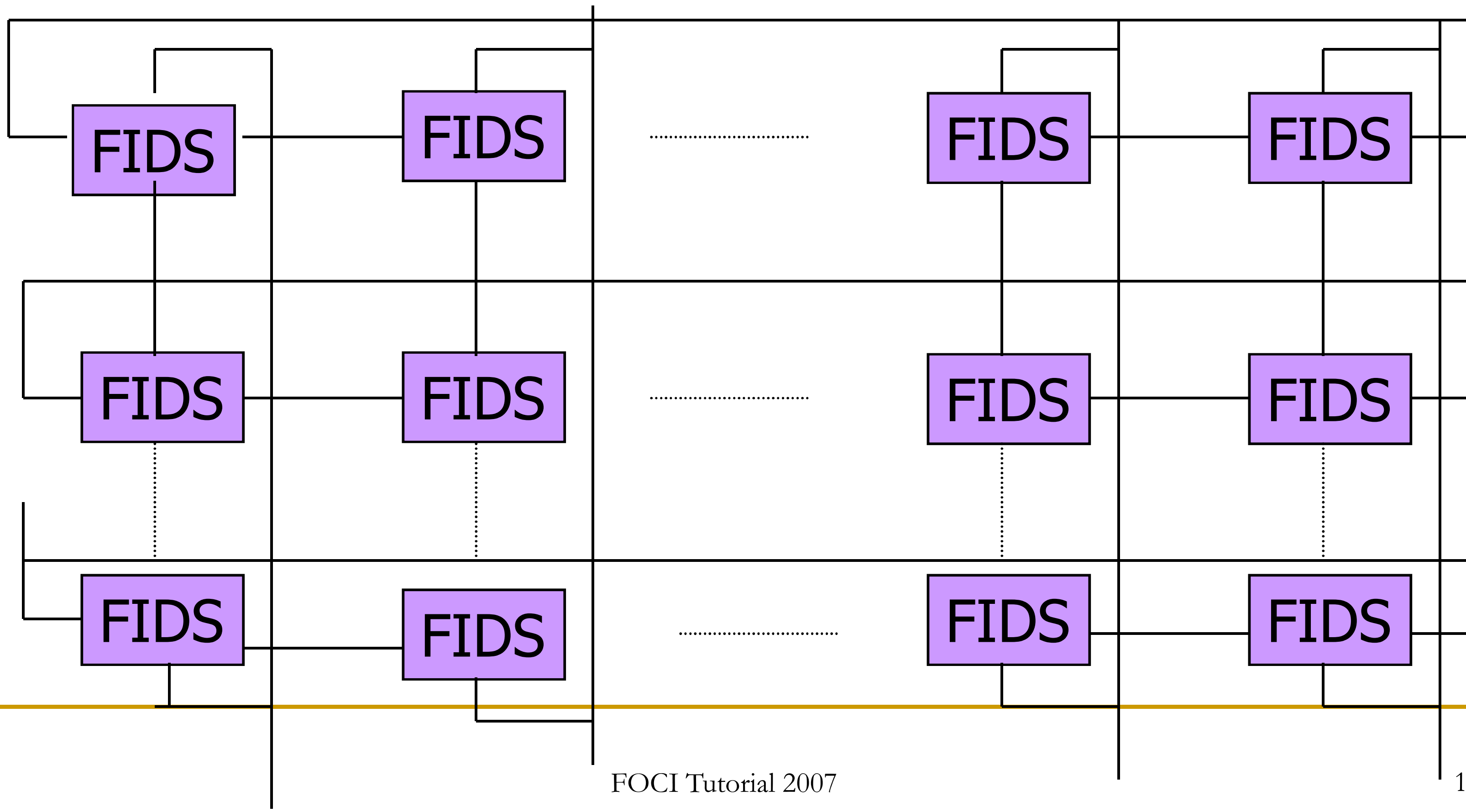
- Apply Cellular automata Concepts:
  - neighborhood topology.
  - Local interaction
  - React to changes in their neighbors.
- Challenging Issues:
  - Secure communication.
  - Synchronization.
  - Remote execution.



Using Cellular Automata - CORAL

J. Jomez 2003

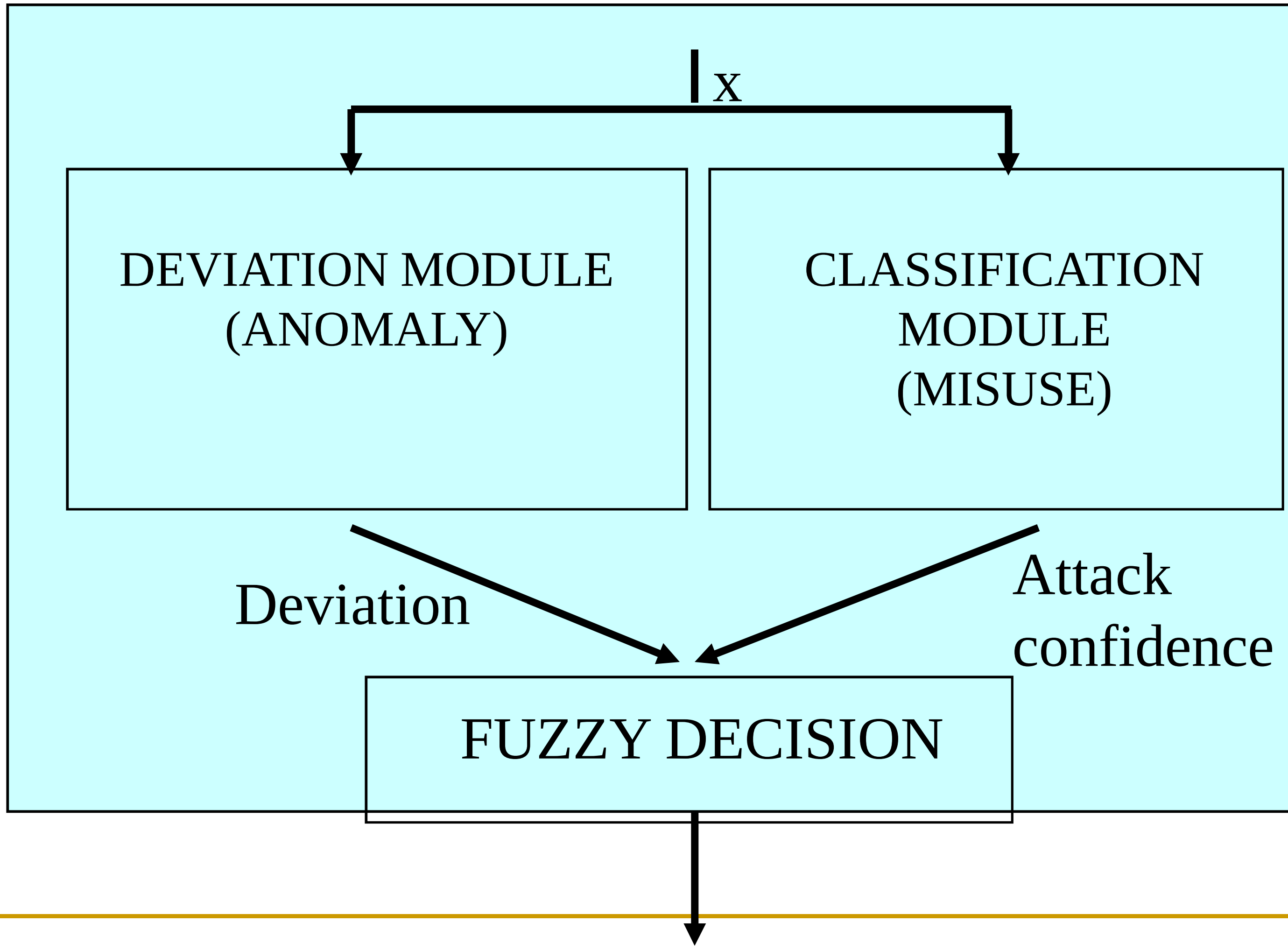
- CORAL : Cell ORganized Attack Lasher



Coral Approach

- The state of a single cell represents the class of a data record presented to it.
- Training:
  - Each cell uses a portion of the training data set
- Decision:
  - The state of the cell automaton will determine the final decision. (Fuzzy Voting)
  - The cell automaton is iterated until a stability criterion is satisfied or maximum number of iters is reached

Fuzzy Integrated Detection System (FIDS)



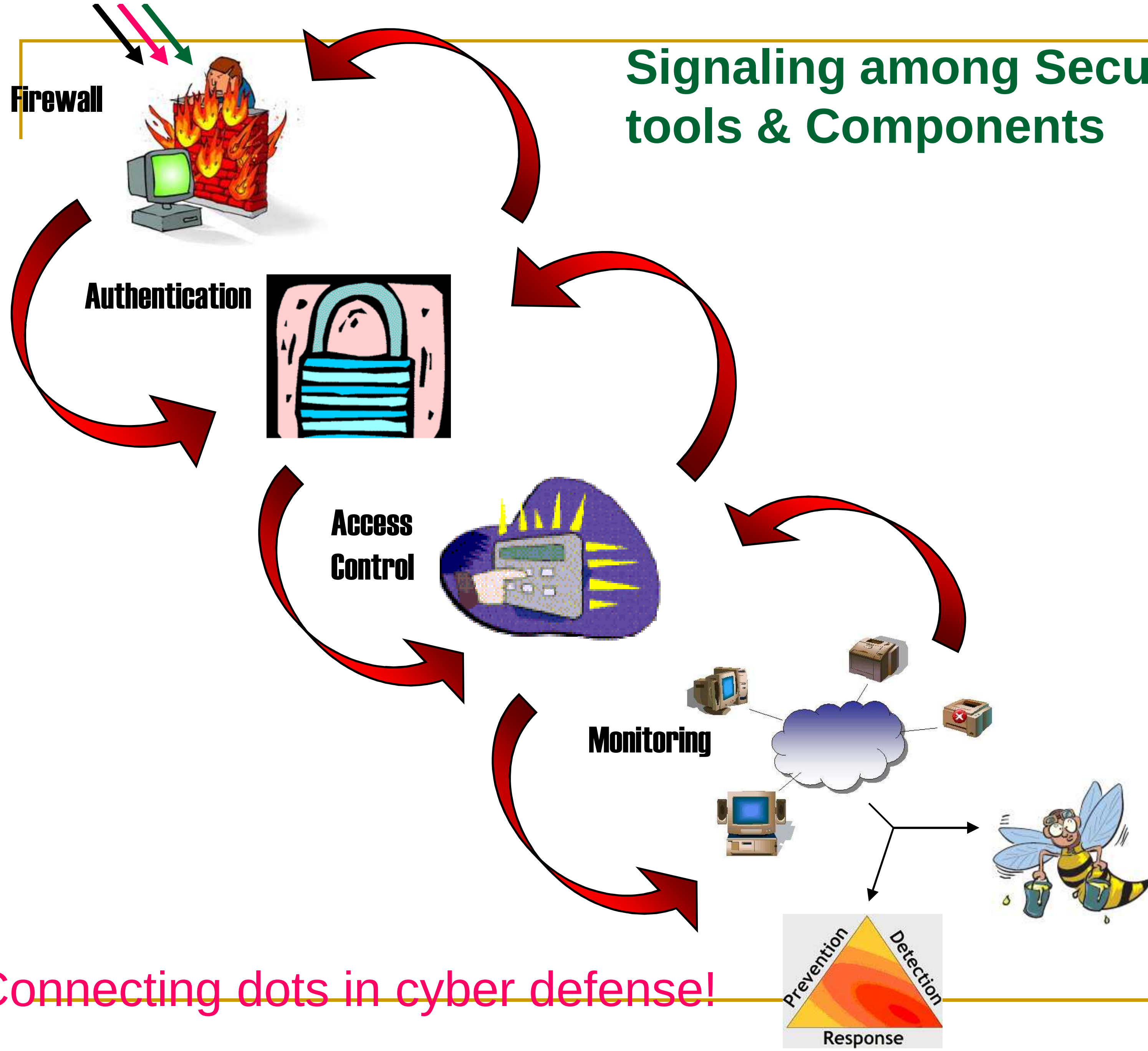
FIDS: Classification Module

- Generated a fuzzy classifier that has a set of fuzzy rules, one per each abnormal class
- The condition part is defined by the monitored parameters and the consequent part is an atomic expression for the classification attribute
- $R_{Abnormal\text{-}1}$: *IF x is MEDIUM and y is HIGH THEN pattern is abnormal*$_1$
- . . .
- $R_{Abnormal\text{-}m}$: *IF x is LOW*
- *THEN pattern is abnormal*$_m$

Major Challenges in Security Agent Technology

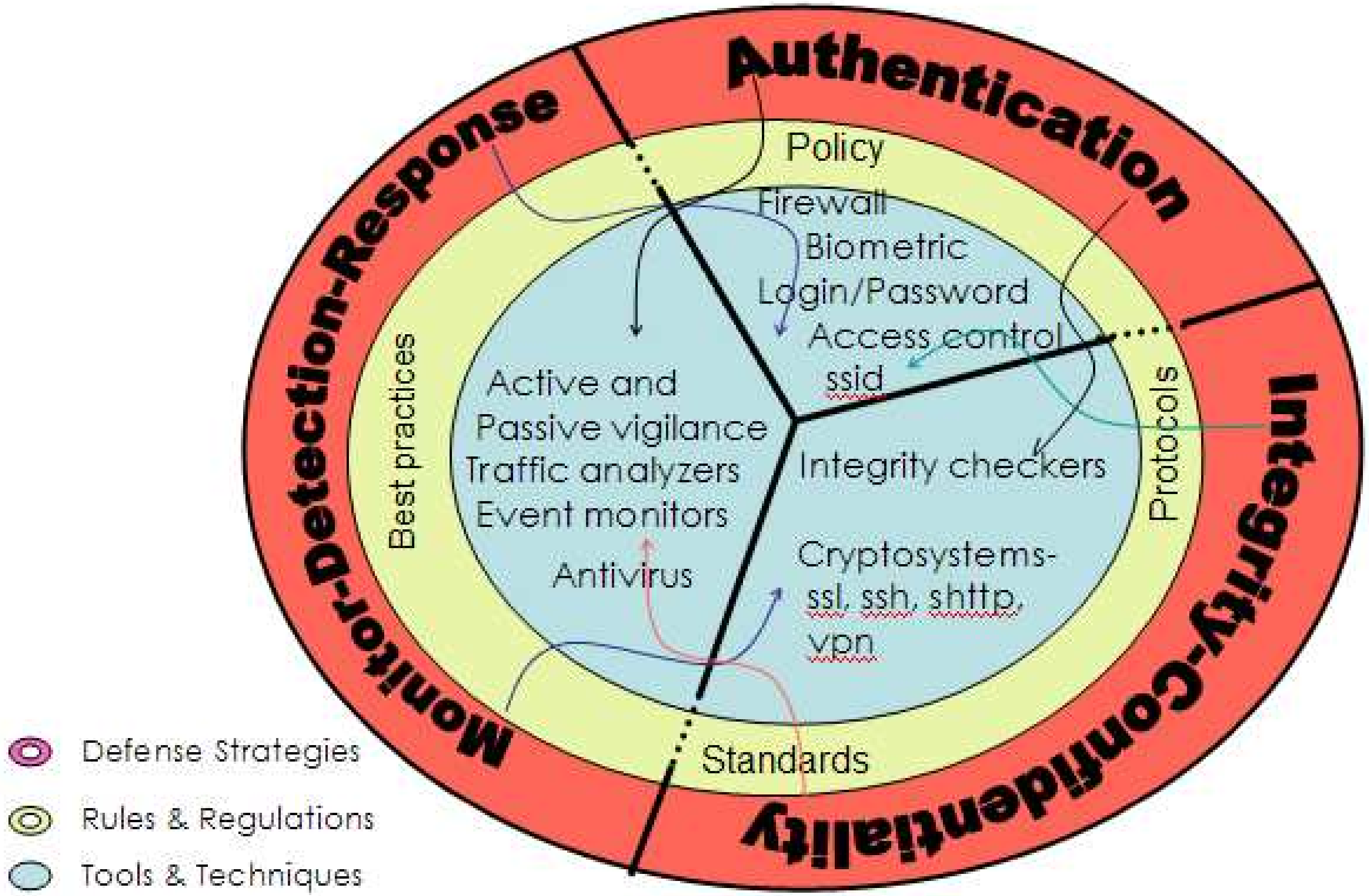
- Integrating various modules
- Automating Agent responses
- Evolving appropriate decision rules
- Prevention of Agent tempering
- Scale up

Signaling among Security tools & Components



Connecting dots in cyber defense!

Cyber Security Management System



Intelligent Security Systems Research Lab (ISSRL)

(http://issrl.cs.memphis.edu)

at

The University of Memphis

- Offering security-related courses
- Developing distributed security agent software (using various Intelligent Techniques) for automated intrusions/anomaly detection and response.

National Cyber Security Research Center

Lab Director:
Prof. Dipankar Dasgupta

Ph.D. Students:
- Joseph Vannucci
- Rodrigo Silva
- Zhou Ji
- Dean Garrett

M.S. Students:
- Sankalp Balachandran
- Jose M Rodriguez
- Rukhsana Azeem
- Serge Salan

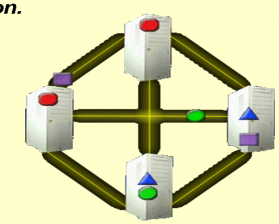
U.G. Students:
- John Clutter
- Mykola Aleshchanov
- Ben Humphreys

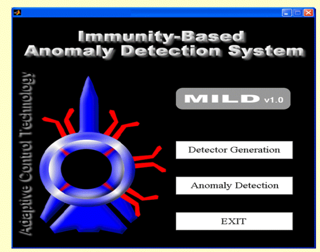
MISSION STATEMENT

We investigate biologically-motivated computing paradigms to design and develop intelligent, adaptive, mobile-agent based security software solutions for protecting computer systems and detecting both known and unknown attack patterns. Our principal research focus is on soft-computing technologies like immunological computation, genetic algorithms and fuzzy neural systems.

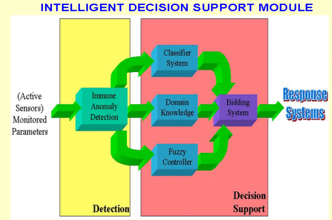
RESEARCH PROJECTS

Our recent cyber security projects include development of Immunity-based Intrusion Detection Systems, Evolving generalized attack detectors, Fuzzy Gravitational Clustering for cyber attack detection.





INTELLIGENT DECISION SUPPORT MODULE



RECENTLY DEVELOPED SOFTWARE

MnRSC: Security Console for Monitoring and Response

SANTA: Security Agent for Network Traffic Analysis

CIDS: Cougaar Based Intrusion Detection System

IAD: Immunity-Based Anomaly Detection

DSS: Distributed Security Scheduler

EXTERNAL RESEARCH SUPPORT

Office of Naval Research (ONR)

Defence Advanced Research Projects Agency (DARPA)

National Science Foundation (NSF)

Lab Web site: http://issrl.cs.memphis.edu/ Homepage: http://www.cs.memphis.edu/~dasgupta/

Some ISSRL Publications

- D. Dasgupta. *Use of Agent Technology for Intrusion Detection*. A chapter in the book A Chapter in the book, Handbook of Information Security, Volume 3, Threats, Vulnerabilities, Prevention, Detection and Management (Part-3),(Editor: Hossein Bidgoli) ISBN: 0-471-64832-9, John Wiley & Sons, Inc., January 2006.
- D. Dasgupta and F. Gonzalez. *Artificial Immune Systems in Intrusion Detection*. A chapter in the book Enhancing Computer Security with Smart Technology," Editor: V. Rao Vemuri, pages 165-208, Auerbach Publications, November 2005.
- D. Dasgupta, F. Gonzalez, K. Yallapu and M. Kaniganti. Multilevel Monitoring and Detection Systems (MMDS). Published in the proceedings of 15th Annual Computer Security Incident Handling Conference (FIRST), Ottawa, Canada, June 22-27, 2003.
- J. Gomez, F. Gonzalez, M. Kaniganti and D. Dasgupta. An Evolutionary Approach to Generate Fuzzy Anomaly Signatures. In the proceedings of the Fourth Annual IEEE Information Assurance Workshop, West Point, NY, June 18-20, 2003.
- J. Gomez, F. Gonzalez and D. Dasgupta, "An Immuno-Fuzzy Approach to Anomaly Detection". In the Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZIEEE), pp.1219-1224, May 25-28, 2003.
- Dipankar Dasgupta and Hal Brian, *Mobile Security Agents for Network Traffic Analysis*, Publication by the IEEE Computer Society Press in the proceedings of the second DARPA Information Survivability Conference and Exposition II (DISCEX-II), 13-14 June 2001 in Anaheim, California.
- Dipankar Dasgupta and Fabio A. Gonzalez, *An Intelligent Decision Support System for Intrusion Detection and Response*, In Lecture Notes in Computer Science (publisher: Springer-Verlag) as the proceedings of International Workshop on Mathematical Methods, Models and Architectures for Computer Networks Security (MMM-ACNS), May 21-23, 2001, St.Petersburg, Russia.
- Dipankar Dasgupta, *Immunity-Based Intrusion Detection Systems: A General Framework*, In the proceedings of the 22nd National Information Systems Security Conference (NISSC), October 18-21, 1999.

References

- R. Heady, G. Luger, A. Maccabe, and M. Sevilla. The Architecture of a Network-level Intrusion Detection System, Technical report, CS90-20. Dept. of Computer Science, University of New Mexico, Albuquerque, NM 87131.
- E. Amoroso, "Intrusion detection", Intrusion.net Books, January 1999.
- J. Allen, A. Christie, W. Fithen, J. McHugh, J. Pickel, and E. Stoner, "State of the practice of intrusion detection technologies", Technical Report CMU/SEI99 -TR-028, ESC-99-028, Carnegie Mellon, Software Engineering Institute, Pittsburgh, Pennsylvania, 1999.
- S. Axelsson, "Intrusion detection systems: A survey and taxonomy", Technical Report No 99-15, Dept. of Computer Engineering, Chalmers University of Technology, Sweden, March 2000.
- J. Sundar, J. Garcia-Fernandez, D. Isaco, E. Spafford, and D. Zamboni, "An architecture for intrusion detection using autonomous agents", Tech. Rep. 98/05, Purdue University, 1998.
- M. Crosbie, "Applying genetic programming to intrusion detection", In Proceedings of the AAAI 1995 Fall Symposium series, November 1995.

References

- W. Lee, S. J. Stolfo, and K. W. Mok, "Mining audit data to build intrusion detection models", Proc. Int. Conf. Knowledge Discovery and Data Mining (KDD'98), pages 66-72, 1998.
- Y. Li, N. Wu, S. Jajosia, and X. S. Wang, "Enhancing profiles for anomaly detection using time granularities", Center for secure information systems. In Journal of Computer Security, 2002.
- S. Bridges and R. Vaughn, “Fuzzy data mining and genetic algorithms applied to intrusion detection”, Proceedings twenty third National Information Security Conference, October 1-19, 2000.
- S. A. Hofmeyr, A. Somayaji, and S. Forrest, "Intrusion detection using sequences of systems call", Journal of Computer Security, 6:151-180, 1998.
- L. A. Zadeh, "Fuzzy sets" in Information and Control, 8: 338-352, 1965
- C. E. Bojarczuk, H. S. Lopes, and A. A. Freitas “Discovering comprehensible classification rules using genetic programming: a case study in medical domain”. Proceedings Genetic and Evolutionary Computation Conference GECCO99, 1999.

References

- W. Fan, W. Lee, M. Miller, S. J. Stolfo, and P. K. Chan, "Using artificial anomalies to detect unknown and know network intrusions", Proceedings of the First IEEE International Conference on Data Mining, 2001.
- K. Yamanishi, Jun-ichi Takeuchi and G. Williams, "On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms", Proceedings of the Sixth ACM International Conference on SIGKDD
- C. Michael & A. Ghosh, "Simple state-based approaches to program-based anomaly detection", to appear in *ACM Transactions on Information and System Security* (TISSEC), 2002
- Fox, Kevin L., Henning, Rhonda R., and Reed, Jonathan H. (1990). A Neural Network Approach Towards Intrusion Detection. In *Proceedings of the 13th National Computer Security Conference.*
- KDD-cup data set. http://kdd.ics.uci.edu/databases/ & kddcup99/kddcup99.html
- J. Gomez, D. Dasgupta and F. Gonzalez, 'Detecting Cyber Attacks with Fuzzy Data Mining Techniques. In the Proceedings of the Third SIAM International Conference on Data Mining, May 1-4, 2003.

References

- H. Debar, M. Becke, & D. Siboni (1992). A Neural Network Component for an Intrusion Detection System. In Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy.
- H. Debar & B. Dorizzi (1992). An Application of a Recurrent Network to an Intrusion Detection System. In Proceedings of the International Joint Conference on Neural Networks. pp. (II)478-483.
- A. K. Ghost et al. (September 27, 1997). “Detecting Anomalous and Unknown Intrusions Against Programs in Real-Time”. DARPA SBIR Phase I Final Report. Reliable Software Technologies.
- K. Tan (1995). The Application of Neural Networks to UNIX Computer Security. In Proceedings of the IEEE International Conference on Neural Networks.
- K.M.C Tan & B.S. Collie(1997). Detection and Classification of TCP/IP Network Services. In Proceedings of the Computer Security Applications Conference.

Questions?

Thank You!