

Overview of UA Ongoing Cybersecurity Projects

Salim Hariri, UA-Site-Director
NSF Cloud and Autonomic Computing Center

hariri@email.arizona.edu

nsfcac.arizona.edu

(520) 621-4378



First Franco-American Workshop
October 17-18, 2013, Lyon France



On Going UA CAC Projects

Supported by: NSF, AFOSR, ARL, AFRL, Intel, IBM, Microsoft, Raytheon, Imaginestics, ISCA Corp, AVIRTEK and Rubio Pharma

- Intrusion Resilient Cloud Services
- AuDIT: Automated Detection of Insider Threat
- 4.2 Million NSF Award for Cybersecurity Scholarship for Service Project at The University of Arizona
- 1.2 Million NSF Award – Hacker Web: Securing Cyber Space: Understanding the Cyber Attackers and Attacks via Social Media Analytics
- Ask CyPert about Cybersecurity Education and Training Programs
- Autonomic Software Protection System (ASPS), and Critical Infrastructure Protection (ACIP) System
 - Smart Buildings and Environments
- Anomaly based Detection of Attacks on Wireless Ad Hoc Networks
- Autonomic Management of Data Center and Cloud Resources
- Autonomic Programming Paradigm



First Franco-American Workshop
October 17-18, 2013, Lyon France



Cyber Security Challenges

- Cyberspace complexity and dynamism make it infeasible for humans to effectively secure and protect
- Current techniques are manual driven, mainly signature base, reactive, and not robust or resilient
- Autonomic Cyber Security (ACS) is a promising paradigm to address current and future cybersecurity challenges

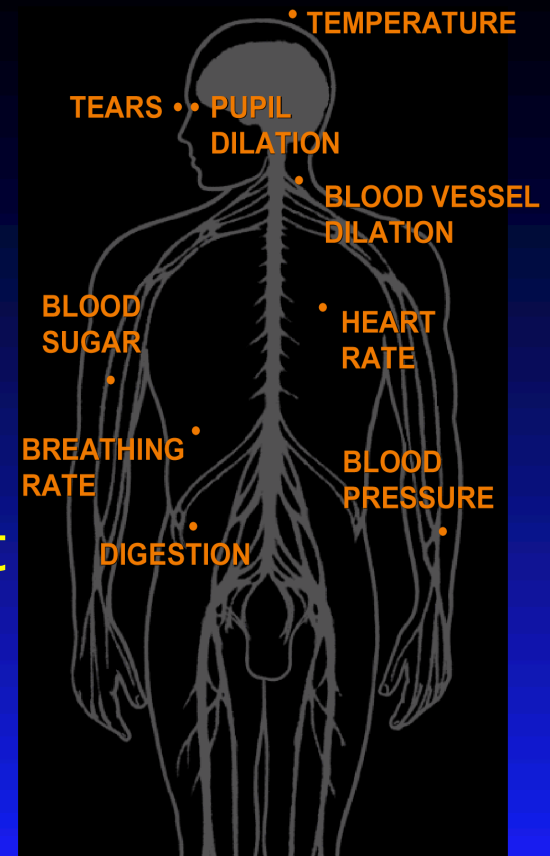
Autonomic Cyber Security (ACS)

Need Biological Like Cyber Nervous System (CNS) that we refer to as ACS.

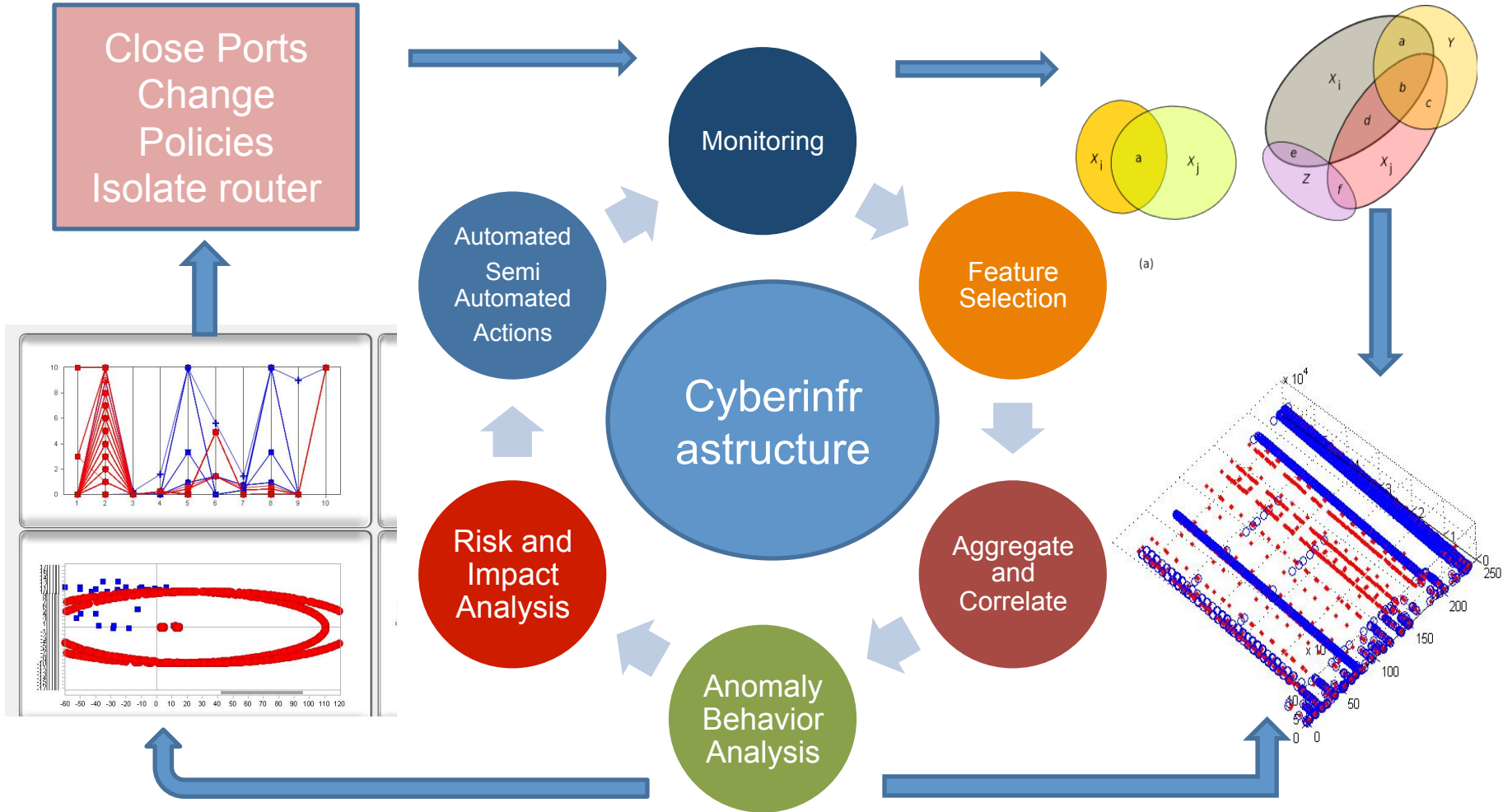
ACS can secure and protect software systems, hardware resources and information services without conscious involvement of users or system administrators

The Autonomic Nervous System Monitors and Regulates:

- without requiring our conscious effort when we run, it increases our heart and breathing rate



ACS Development Methodology



ACS Capabilities

- 🐱 Developing an innovative technology to build Autonomic Cyber Security (ACS) with capabilities ***similar to the human nervous system***,
 - Software systems, computers, and networks that can ***self-manage and proactively protect themselves in real-time with little or no involvement of users or system administrators***.
 - These systems just focus on functions they provide while the ACS performs what is necessary to self-protect their operations and services.

ACS Key Components

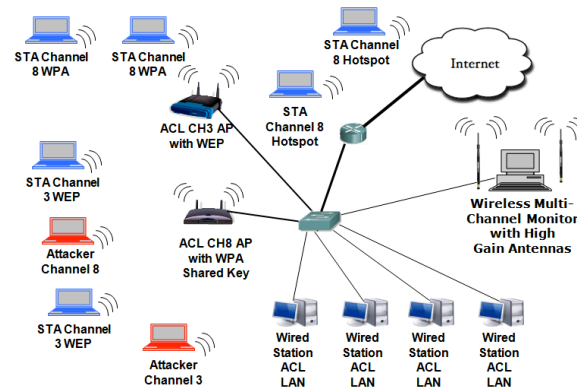
- 🐱 Automated and Integrated Management (AIM) Methodology
- 🐱 Appflow: A data structure that captures the current state of the system
- 🐱 Anomaly Behavior Analysis (ABA) Methodology – low false alarms, and successfully implemented to TCP, UDP, IP, MAC, DNS, HTTP, WiFi, Modbus, etc.
- 🐱 Self-Management: It is a software engine to provide automated and adaptive management services for hardware/software resources
- 🐱 Software Behavior Encryption (SBE)
 - Based on Moving Target Defense (MTD) technique

CAC Cybersecurity Test-beds

Smart Grid



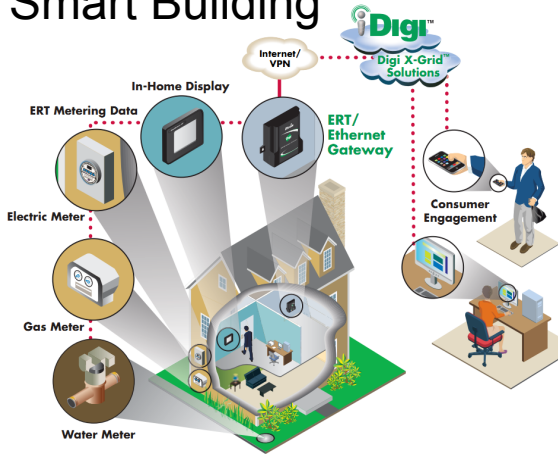
Wireless Test-bed



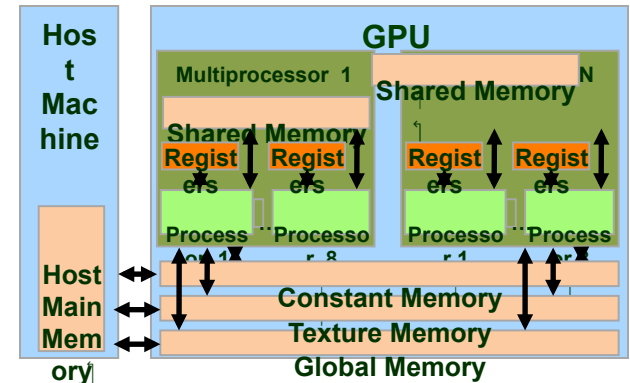
Private Cloud



Smart Building



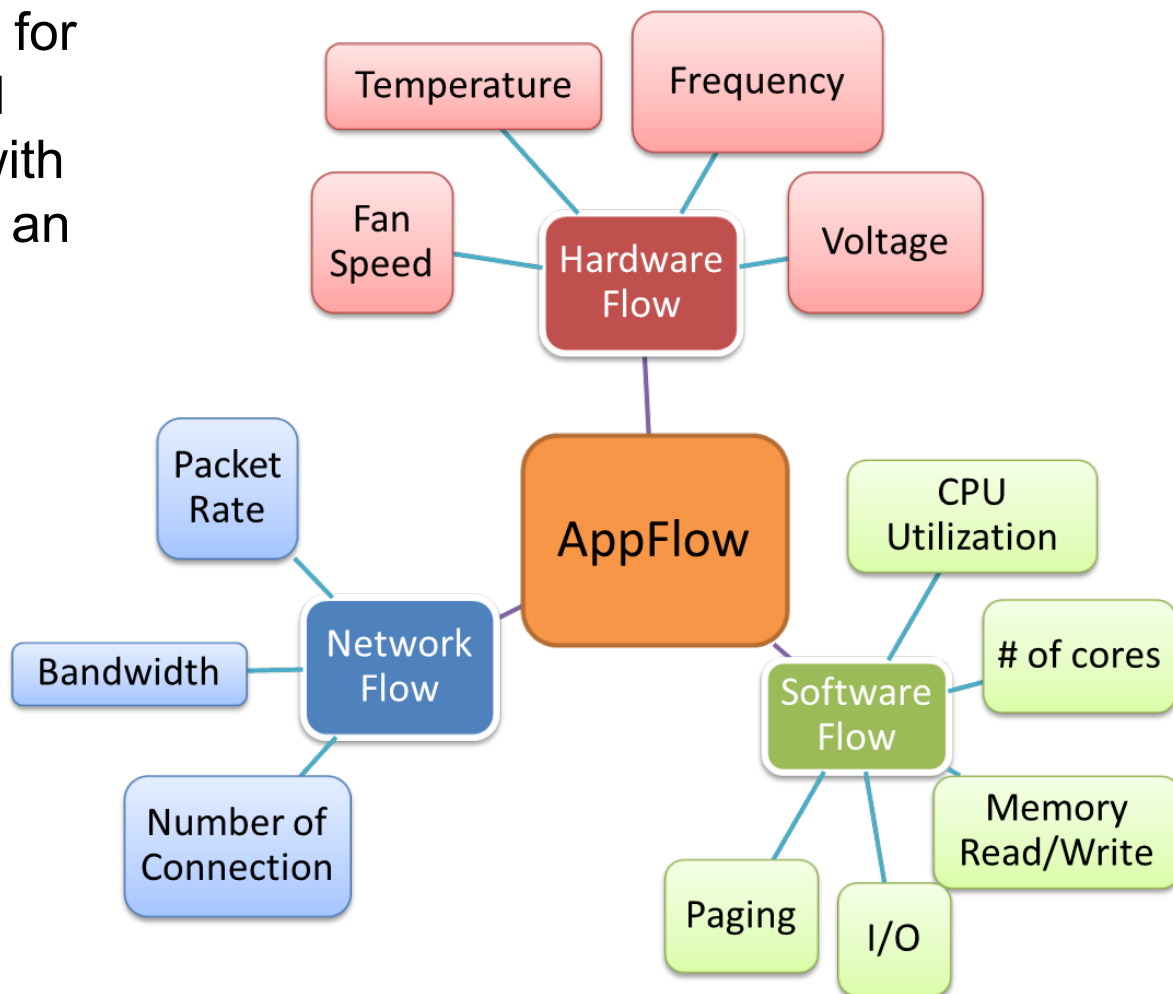
GPU Cluster



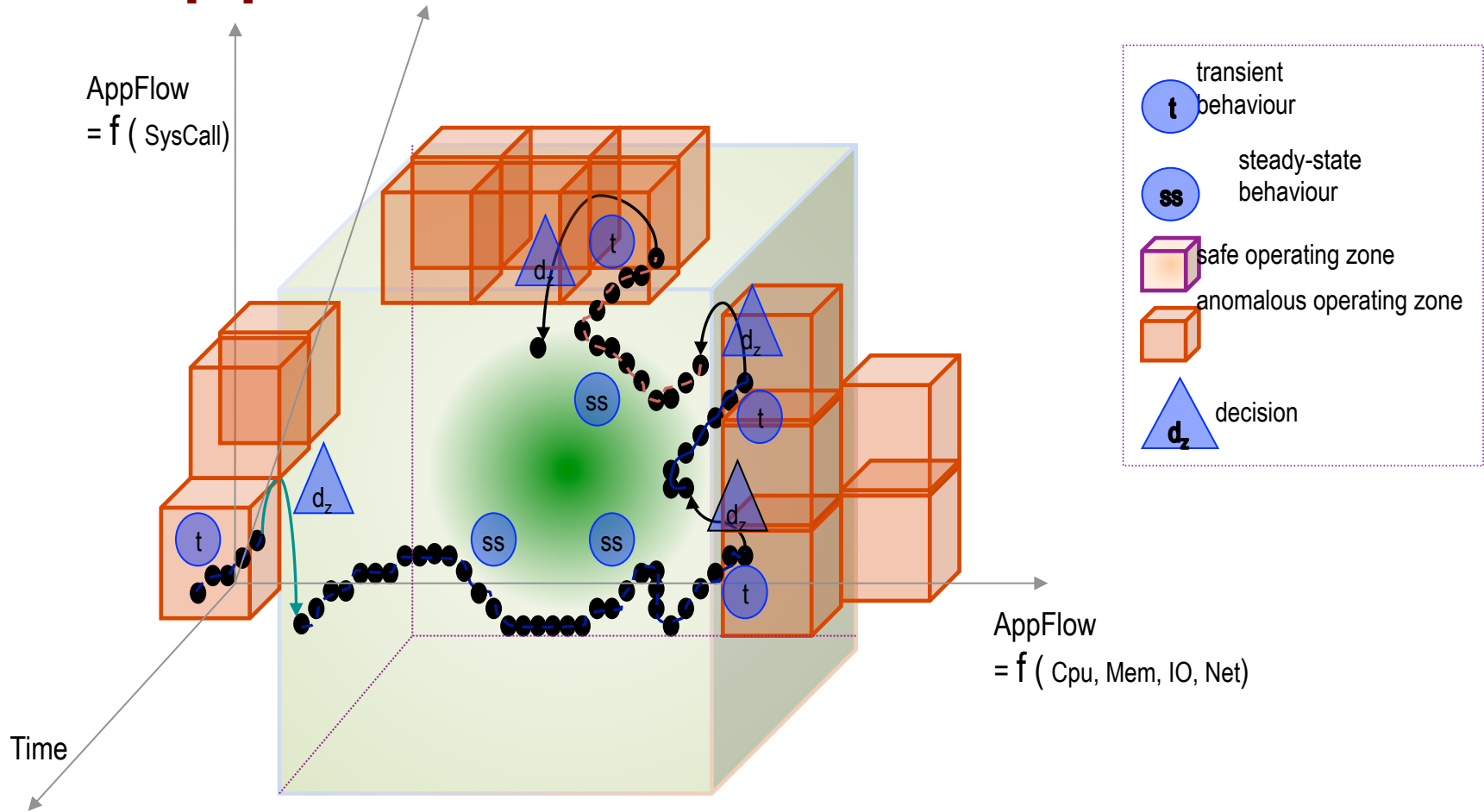
Application Flow (Appflow)

A data structure used for holding the monitored features associated with all resources used by an application at runtime

Similar to the biological measurements (heart rate, body temperature, blood pressure, cholesterol, etc.)

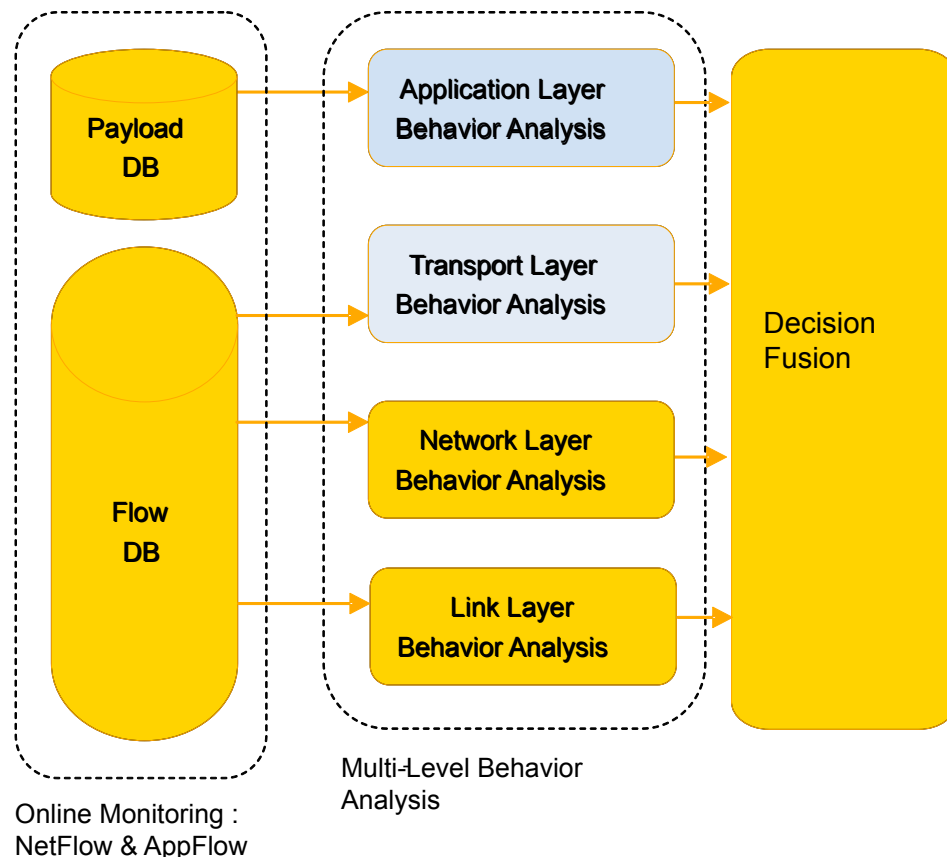


AppFlow Behavior at Runtime

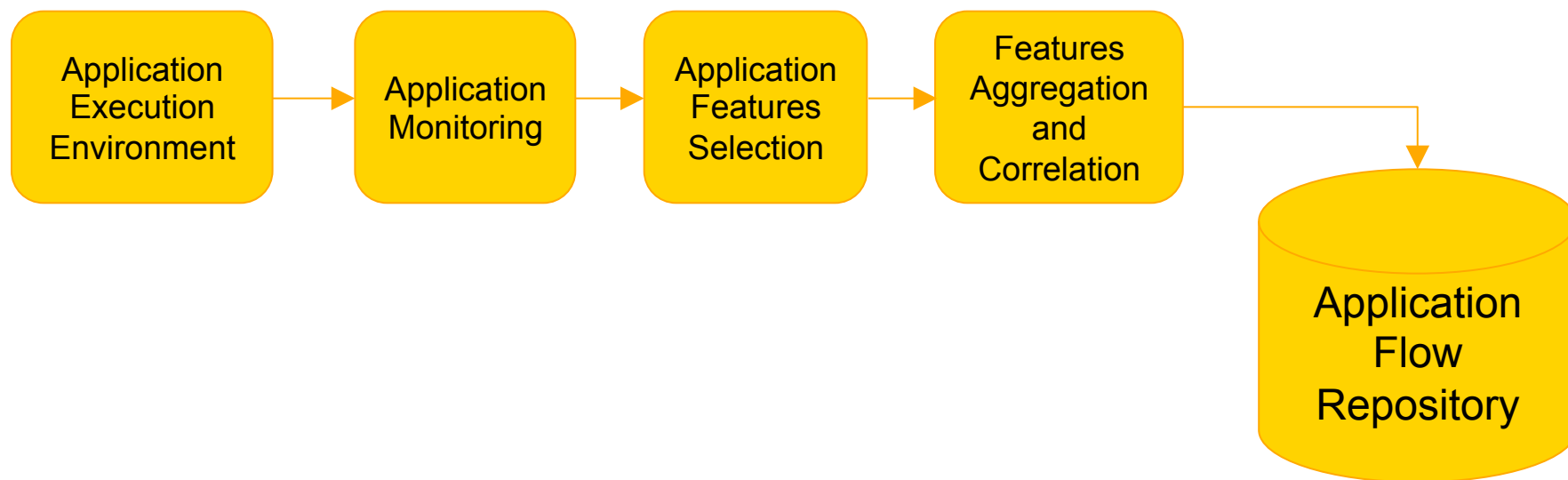


Anomaly Behavior Analysis (ABA)

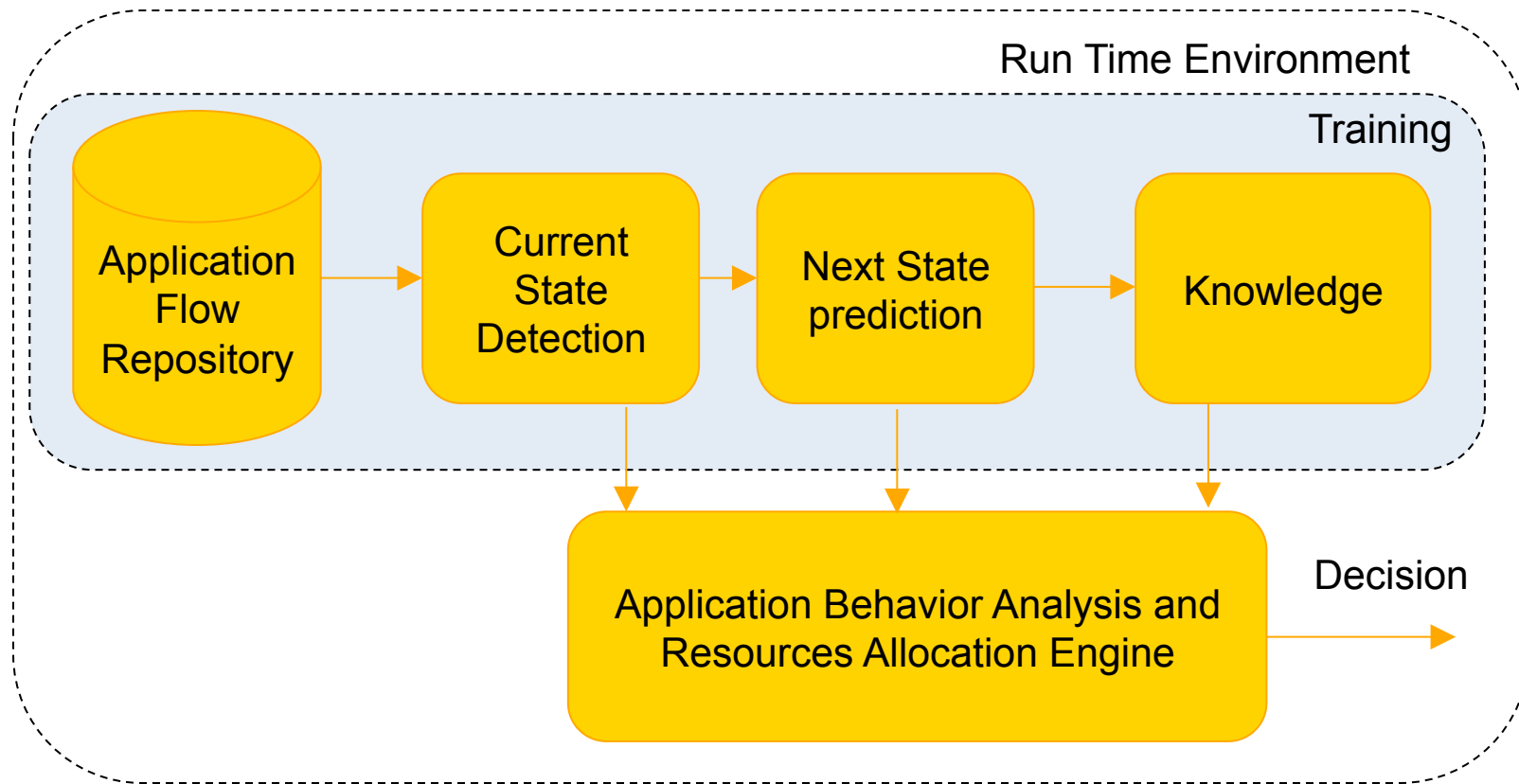
- 🐾 ABA performs fine-grained behavior analysis of applications, software systems, and protocols to determine whether they are operating normally or not
- 🐾 The only assumption it makes that we know how the analyzed component behaves when it is operating normally
- 🐾 This allows us to detect any unknown attacks (zero attack detection)



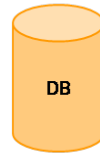
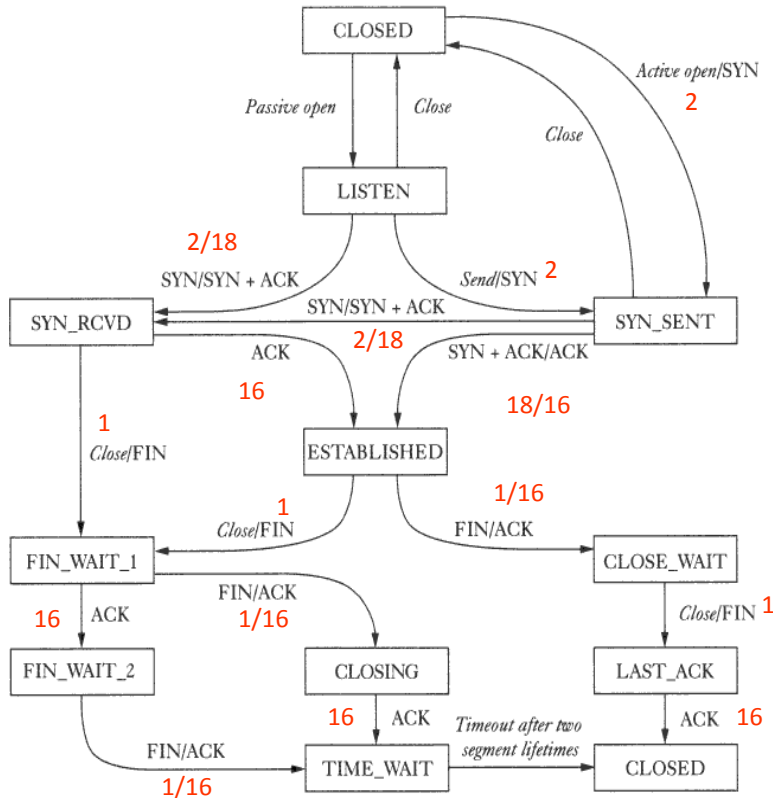
Application Behavior Analysis: AppFlow based Methodology



Application Behavior Analysis

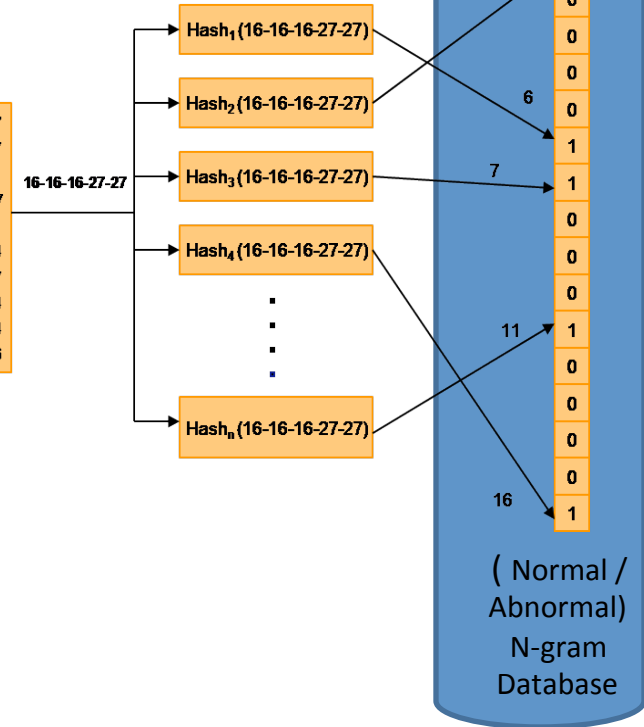


TCP Behavior Analysis

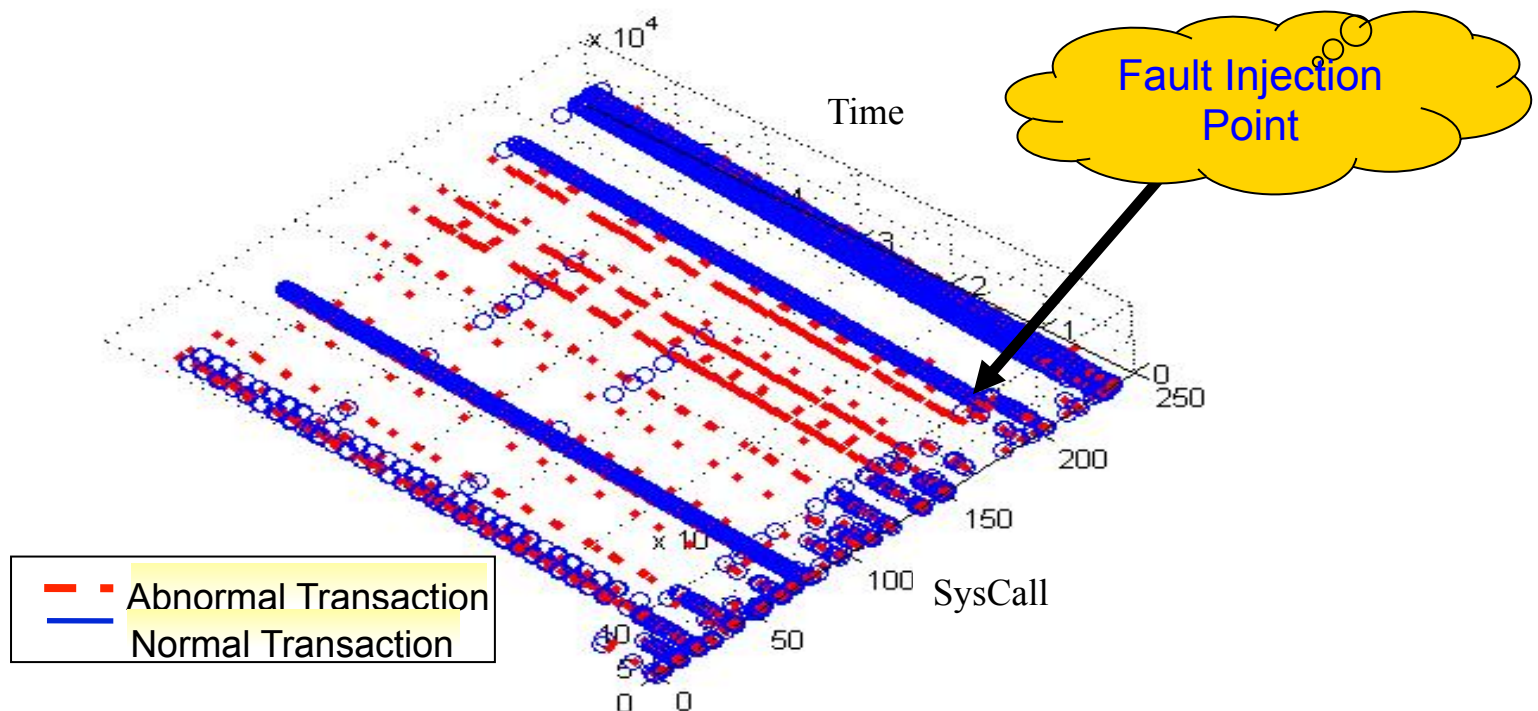


16-16-16-27-27
 16-76-13-27-97
 32-15-18-46-41
 16-55-16-53-27
 12-12-18-41-41
 18-16-16-25-24
 16-85-16-27-27
 24-24-24-16-24
 22-24-23-16-24
 26-26-25-25-26

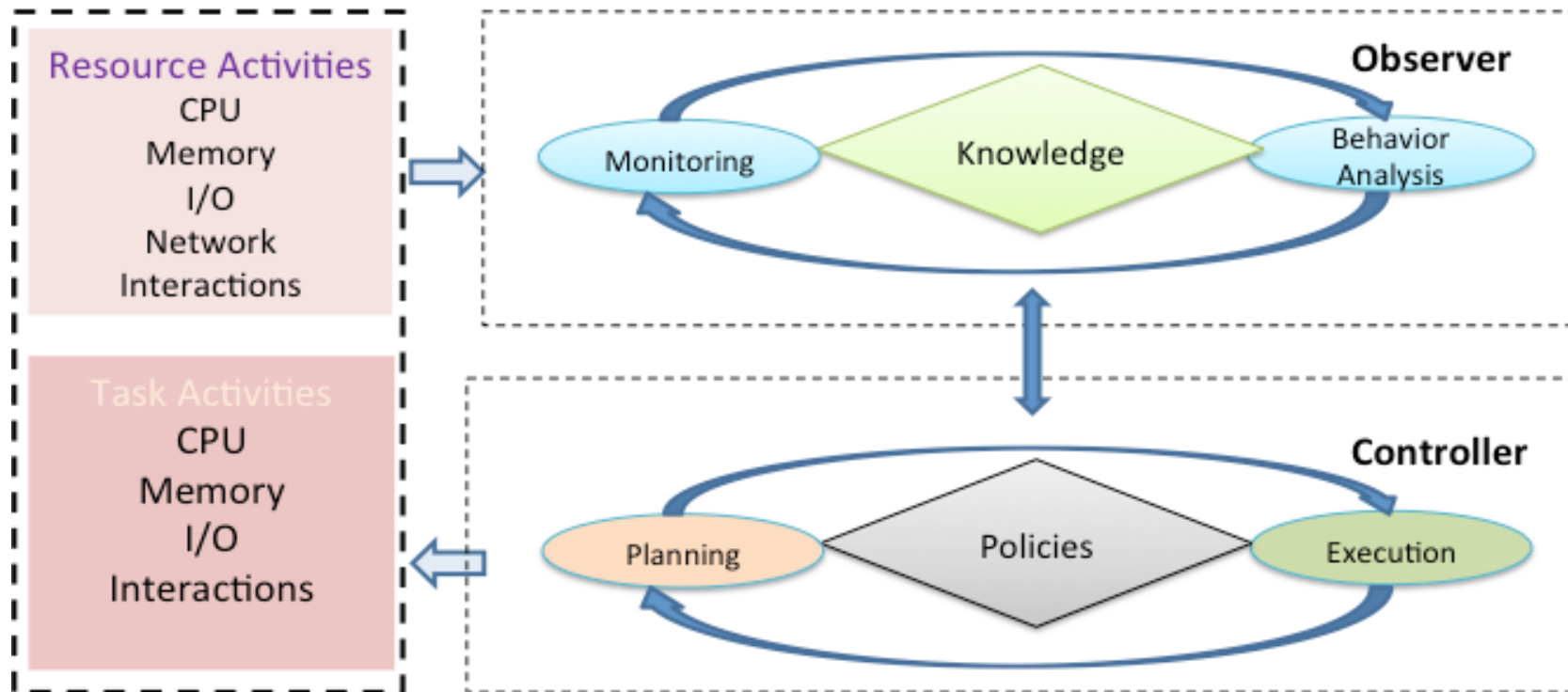
Training



Statistical Distribution of System Calls (Normal vs Abnormal)



Automated and Integrated Management (AIM) Engine



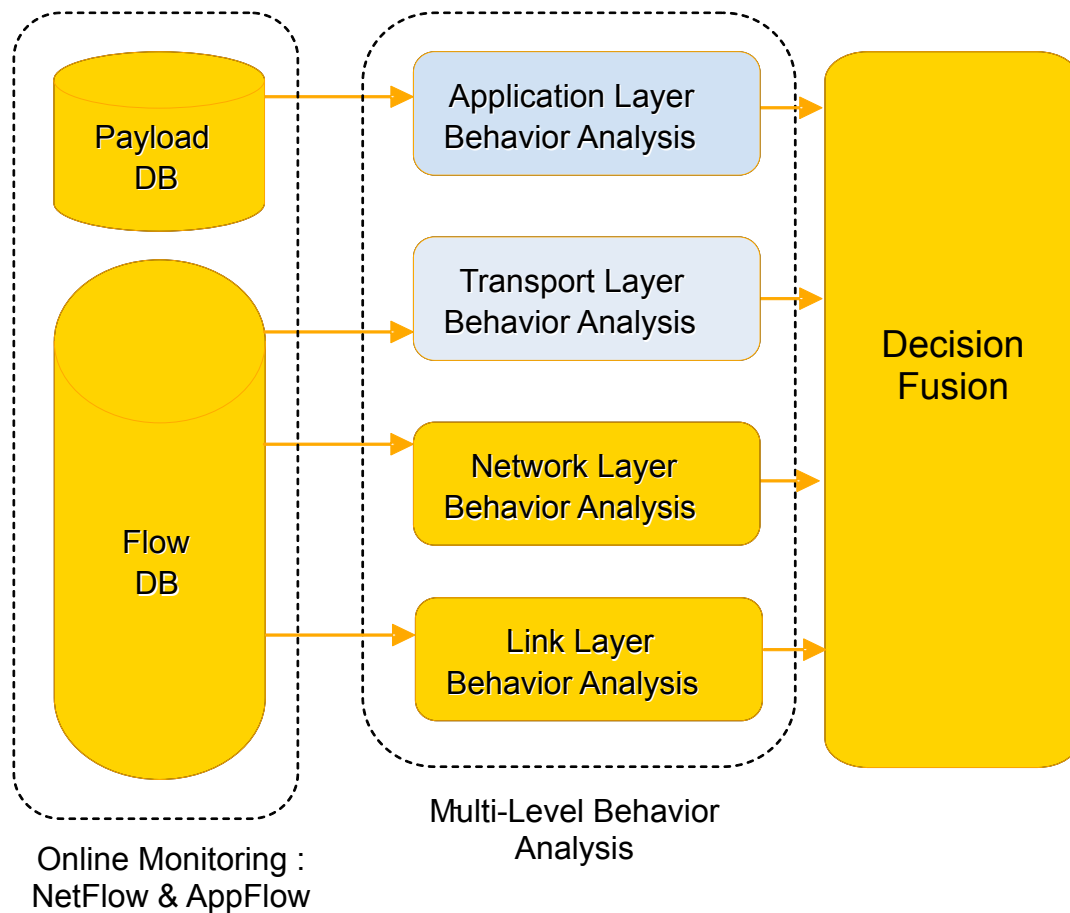
ANOMALY BEHAVIOR ANALYSIS (ABA) OF DNS PROTOCOL



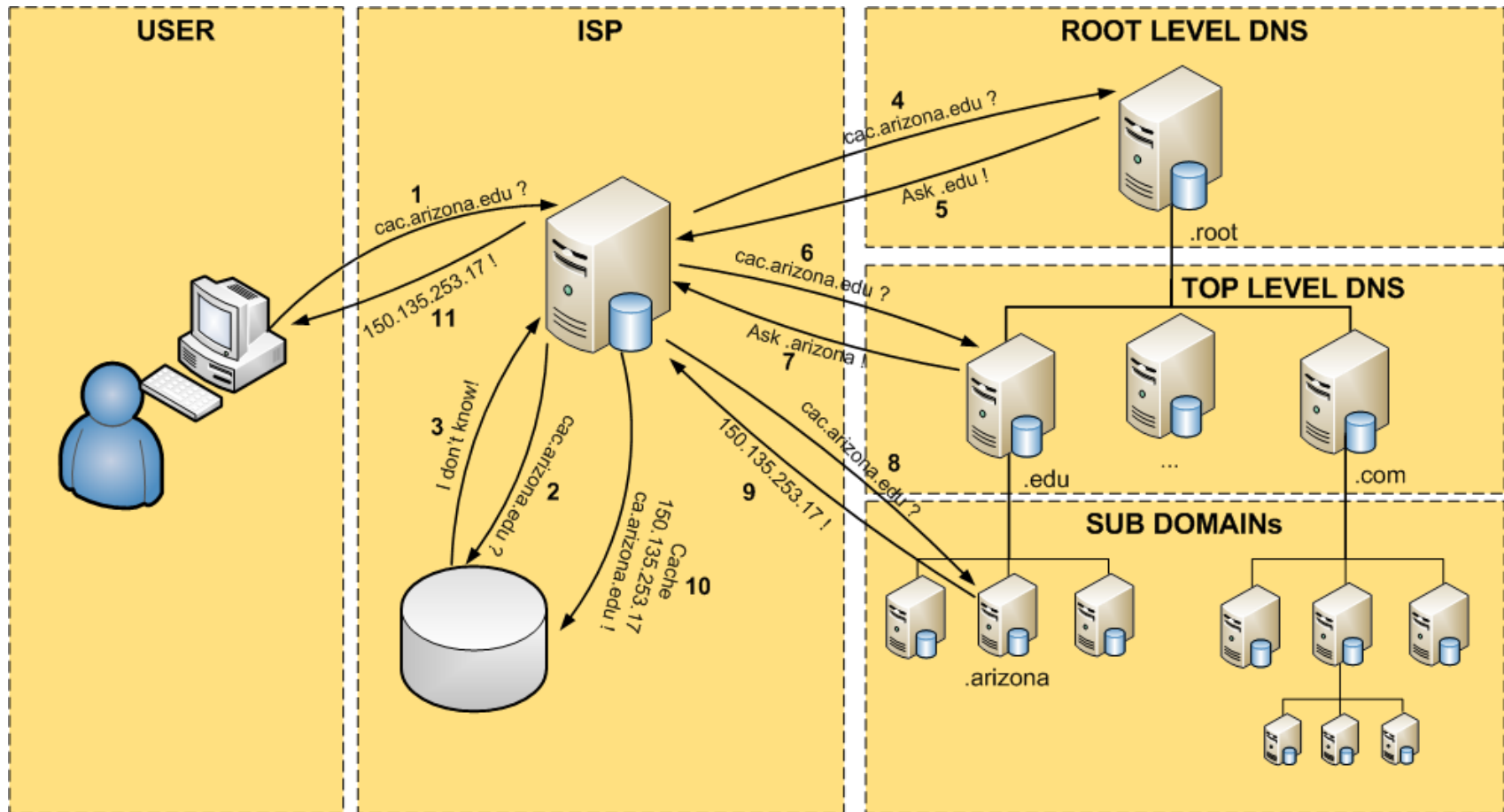
First Franco-American Workshop
October 17-18, 2013, Lyon France









Anomaly Behavior Analysis (ABA) Methodology

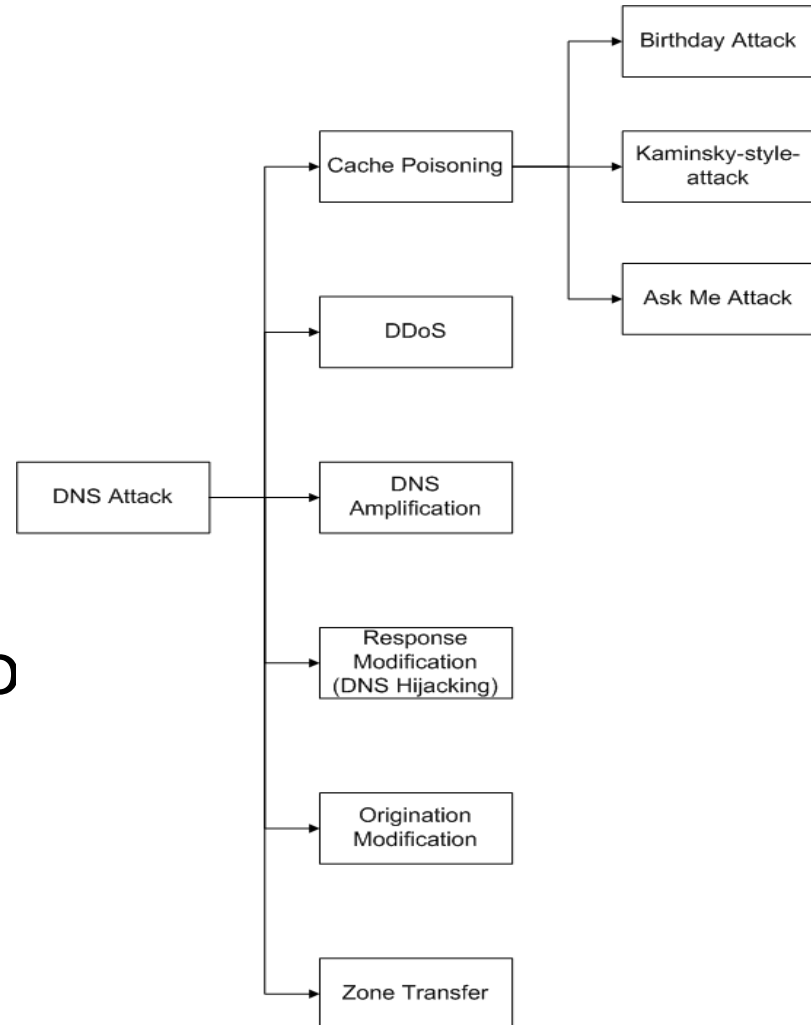


DNS Behavior Analysis Unit

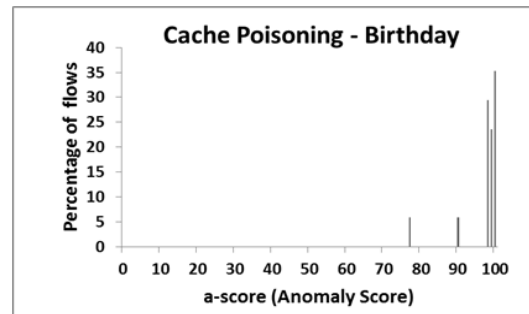
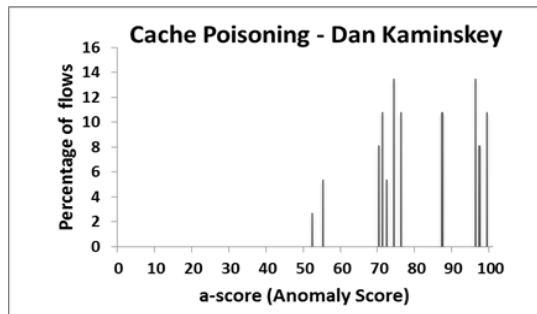
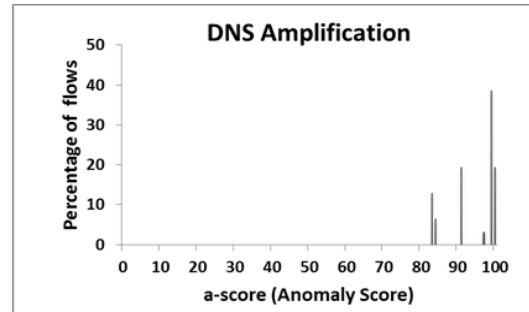
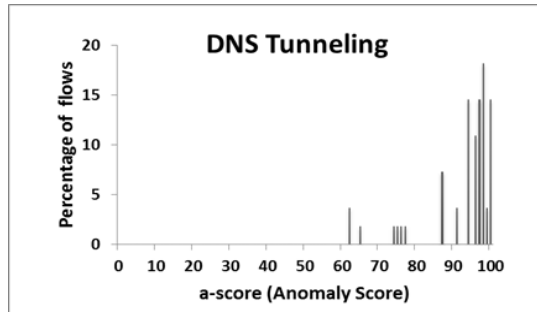
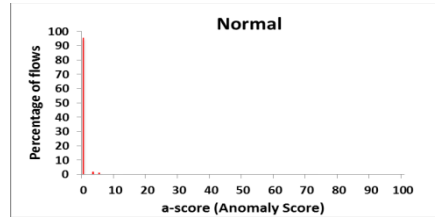


DNS Attacks

-  Cache Poisoning
-  DNS Hijacking
-  DNS Amplification
-  DDoS
-  Origination Modification
-  Zone Transfer

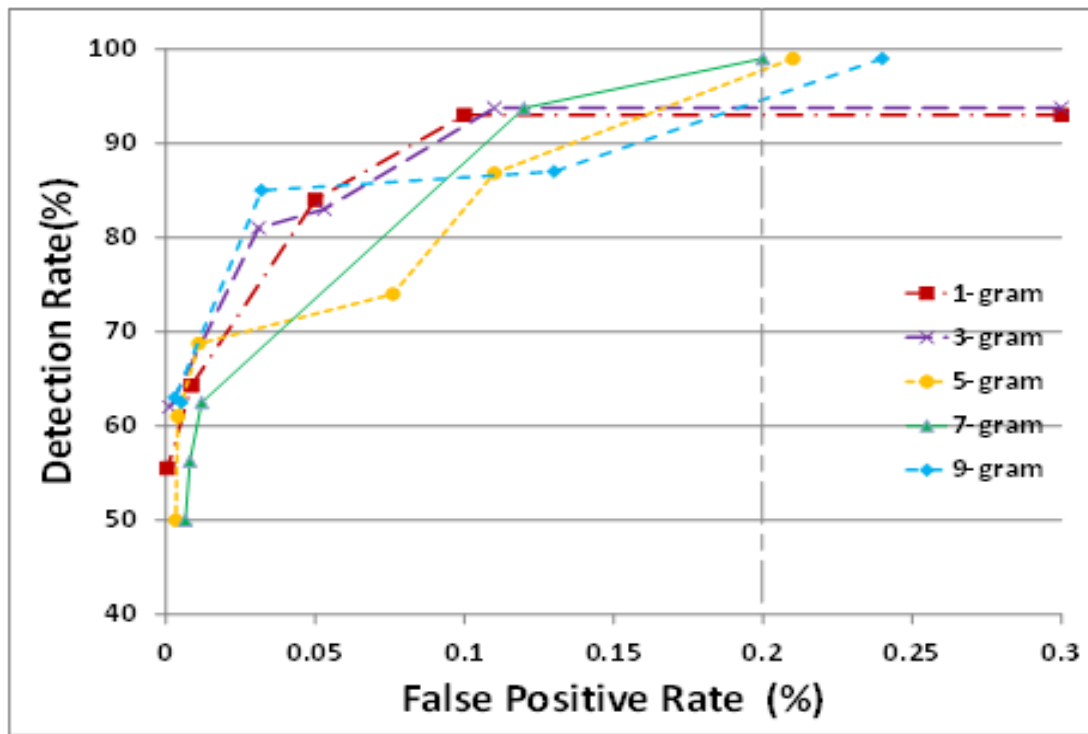


DNS BAU Results



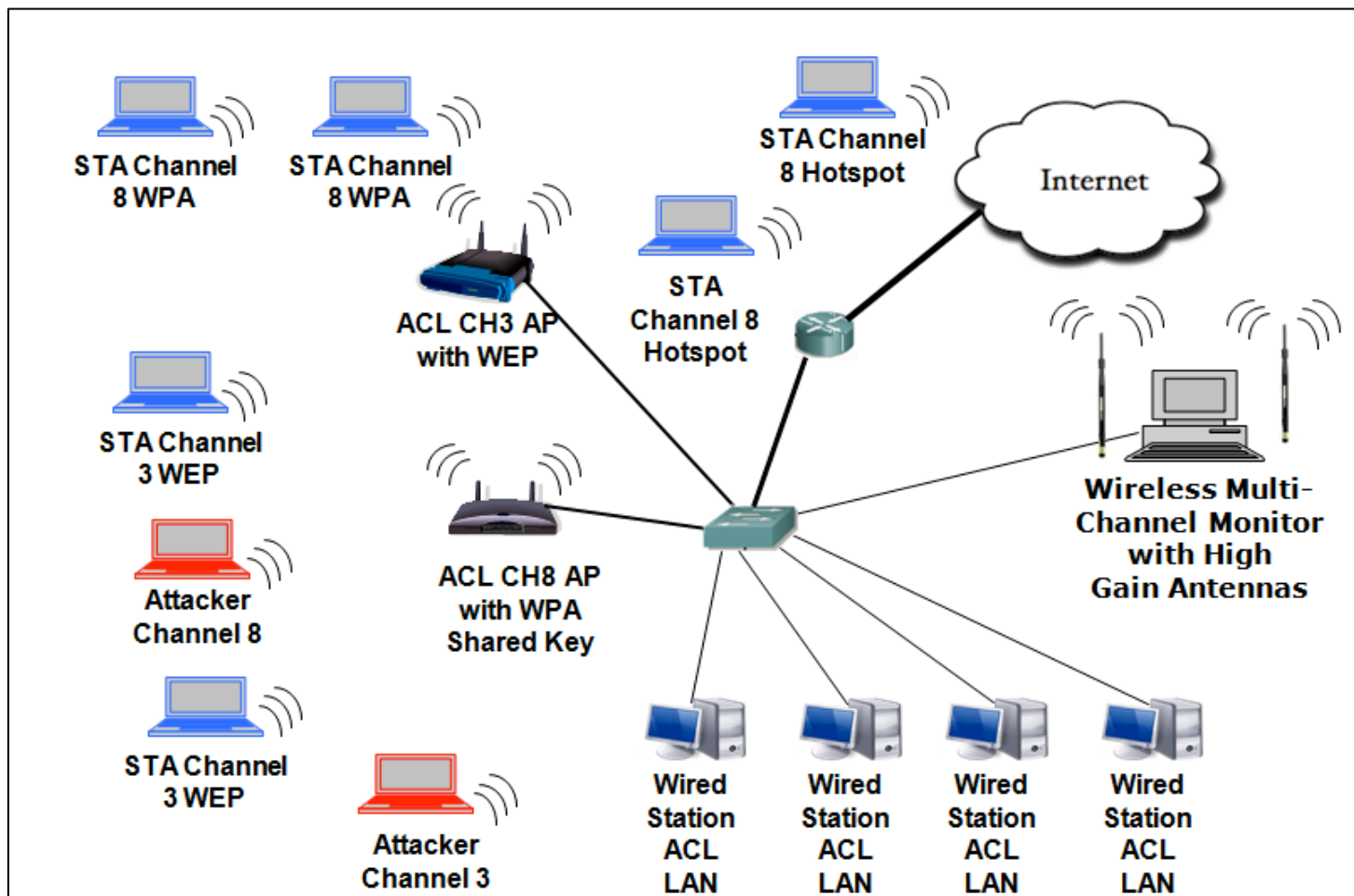
The anomaly score distribution for different type of attack traffic

DNS Results



ROC (Receiver Operating Characteristics) for different n-gram sizes.

ABA for WiFi (802.11) Protocol



Wireless Flow Key Analysis

$$n(\text{ngram}_i) = \min(\text{count}(\text{ngram}_i), \text{moc}(\text{ngram}_i))$$

$$np(\text{flow}) = \sum_{\text{ngram}_i \in \text{flow}} n(\text{ngram}_i)$$

$$allp(\text{flow}) = \sum_{\text{ngram}_i \in \text{flow}} \text{count}(\text{ngram}_i)$$

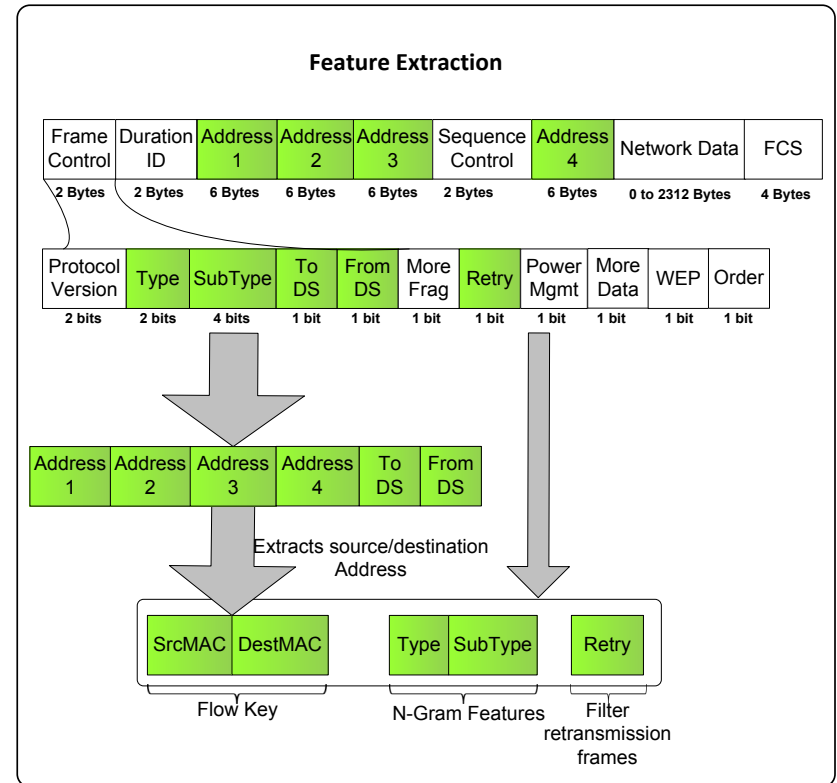
$$\text{score}(\text{flow}) = \frac{(1 - np(\text{flow}))}{allp(\text{flow})} \times 100$$

count (ngram_i): frequency of the ngram_i in the flow

moc (ngram_i): maximum observed count for ngram_i during training

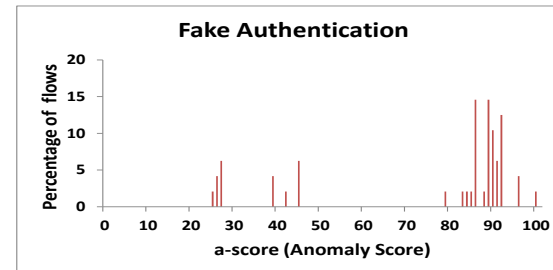
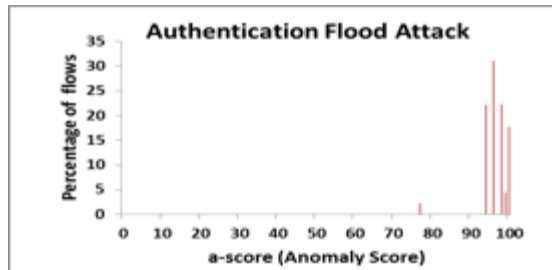
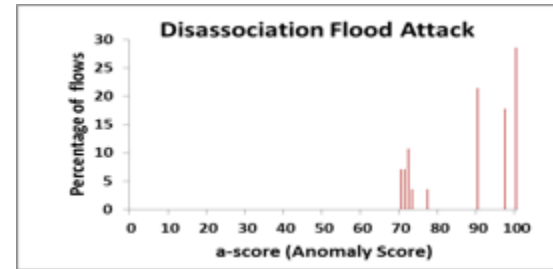
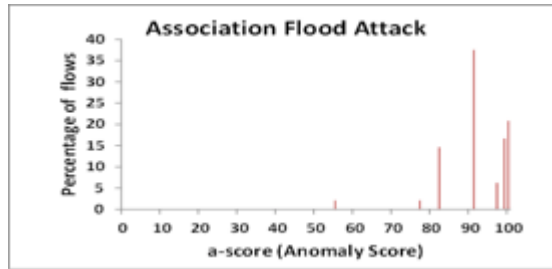
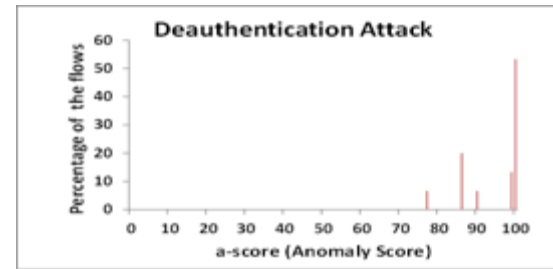
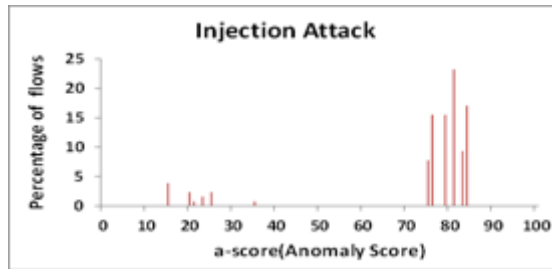
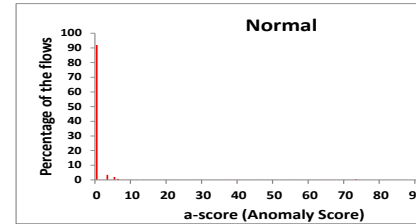
np (flow): number of normal n-gram patterns in the flow

allp (flow): number of all observed patterns in that flow



Experimental Results and Evaluation

- We collected around 216 million frames
- For 4-grams, we observed 922 unique patterns from analyzing 102 Million frames



AuDIT: Automated Detection of Insider Threat



First Franco-American Workshop
October 17-18, 2013, Lyon France



Insider Threats

- A current or former employee or business partner who has authorized access to an organization's resources and intentionally misused that access (CERT, 2012)
- Cited as one of the greatest security threats to organizations (e.g., Boss et al. 2009; Holmlund et al. 2011)
 - 46% of security breaches are caused by insiders (U.S. Secret Service, 2010)
 - Costs “tens, if not hundreds of billions of dollars” (United Nations, 2005, p. xxiii)
 - Takes an average of 416 days to detect a breach (HP Cyber Risk Report, 2012)
- Three proposed solutions: AuDIT, CAT, and ADMIT

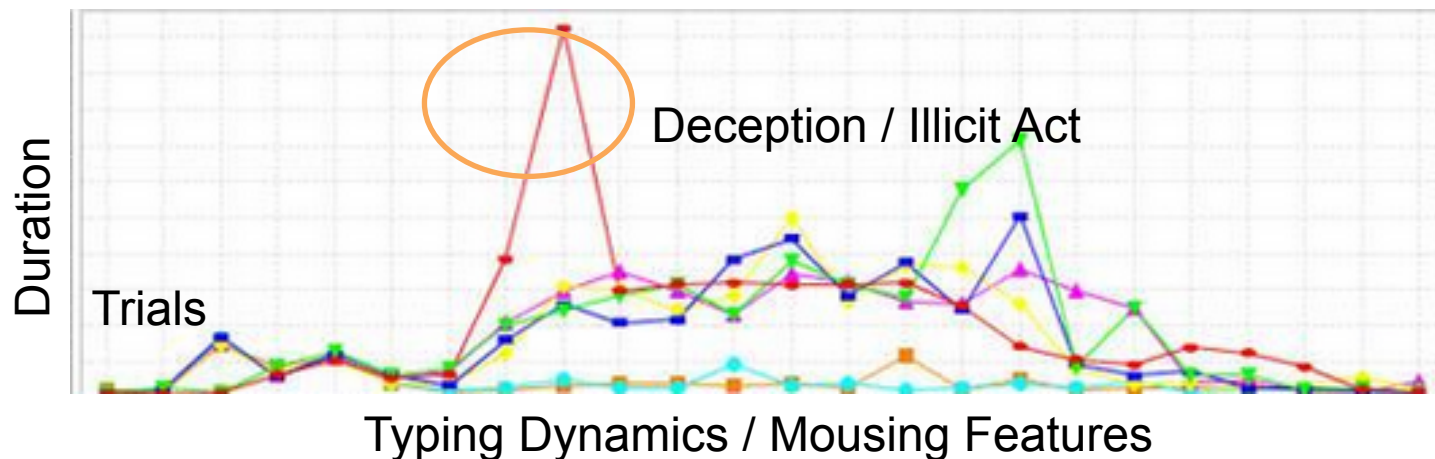


Detecting Insider Threats

- Examples
 - Polygraph
 - Log analysis
 - Investigation surveys
 - Pre-employment screening surveys
- Shortcomings
 - Polygraphs are expensive, time consuming, not always legal, not scalable
 - Log analysis is time consuming and post-hoc
 - Insider threats can lie in surveys

Insider Threats

- Illicit activities cause a stage change in individuals (e.g., heightened emotion, stress, etc.)
- Establish a 'normal' baseline of individuals
- Detects anomalies through mouse / keystroke analysis and system usage characteristics
 - E.g., detect that someone is experiencing heightened emotion while copying a file from a sensitive directory



AuDIT: Automated Detection of Insider Threat

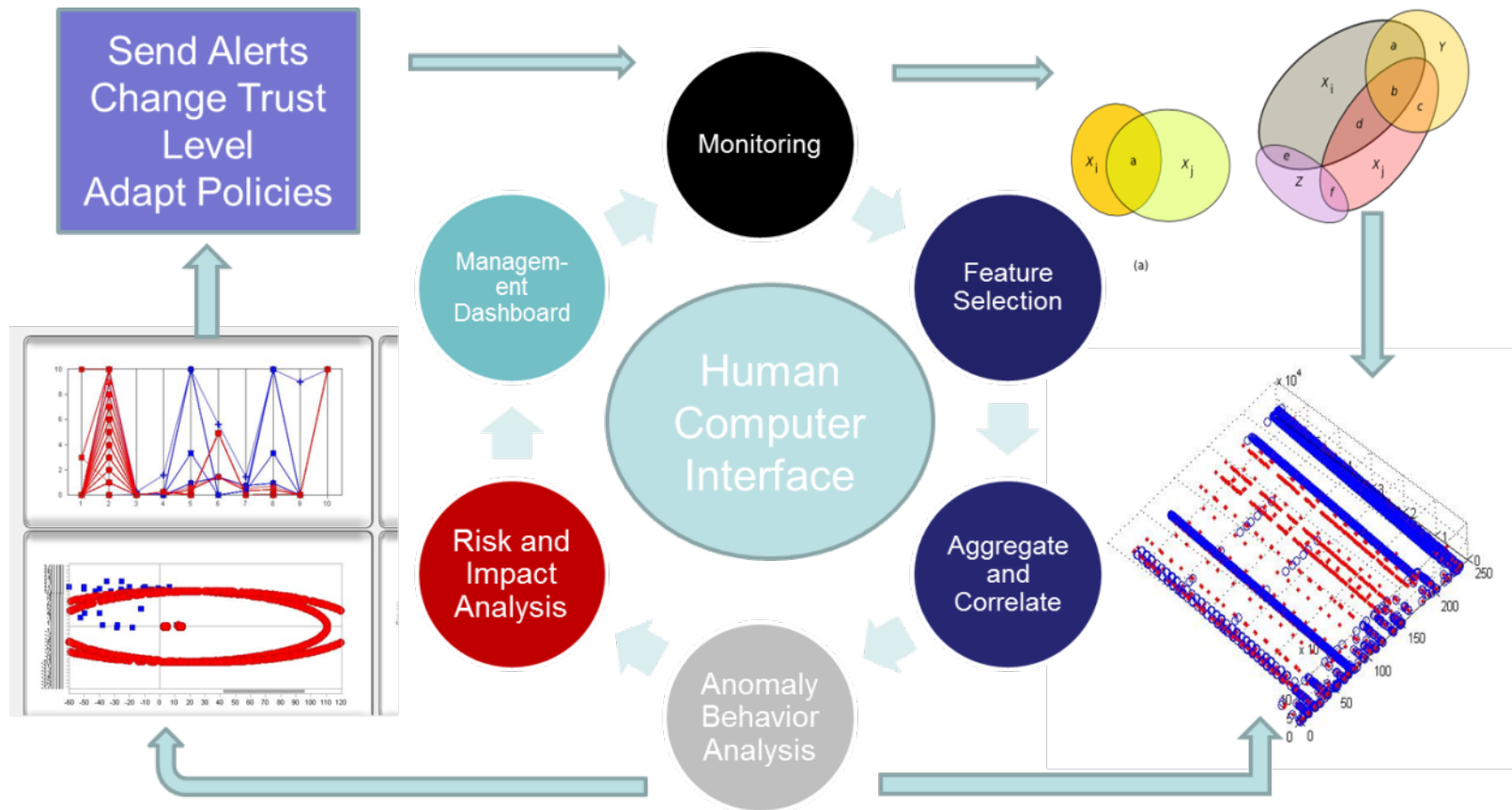
Integrates human behavioral monitoring via non-invasive mouse and keyboard usage patterns with system resource usage for detecting anomalies that could be indicative of insider threats.



AuDIT Methodology

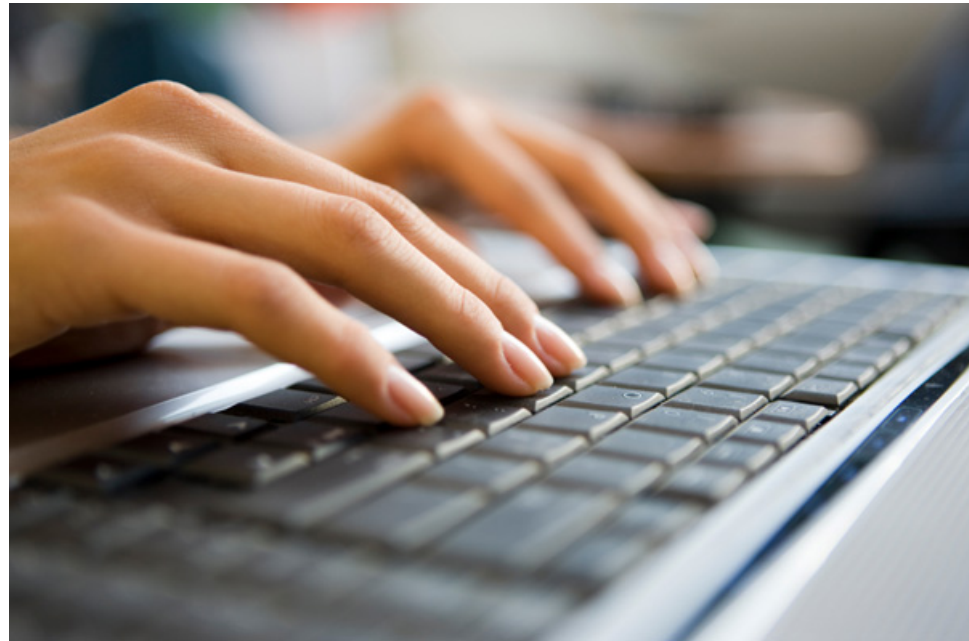
- Establish a ‘normal’ baseline of individuals for resource usage and mouse / keyboard usage
- Detect anomalies through continuous monitoring of mouse / keystroke use and system resource use
 - E.g., detect that someone is experiencing heightened emotion while copying a file from a sensitive directory that they have never done before

AuDIT Framework

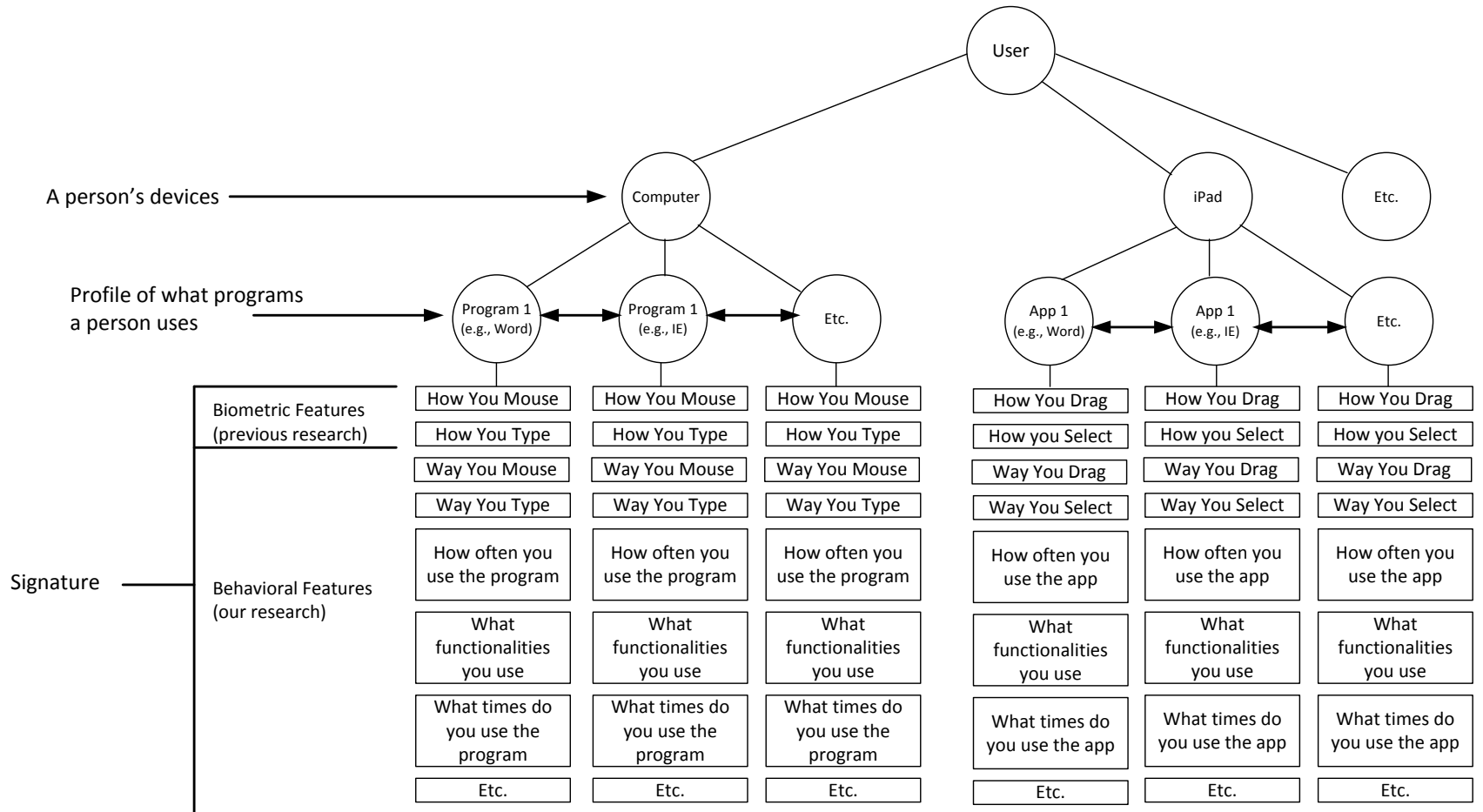


CAT: Continuous Authentication Tool

- Insider threats:
 - Steal credentials (user names, passwords, etc.)
 - Access unlocked computers
 - Disguise identity
- People have a unique mousing and typing signature
 - How / way you mouse
 - How / way you type
- People have unique system usage patterns
 - Time of day, applications, etc.



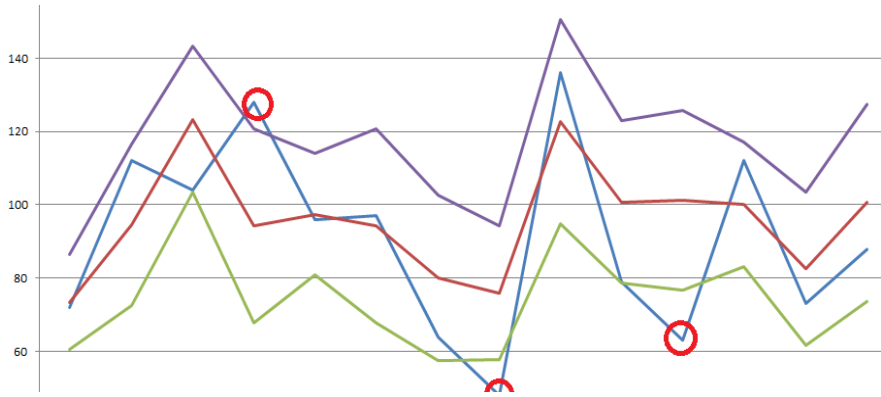
Cyber-Social Behavior Metrics



Building a Cyber DNA

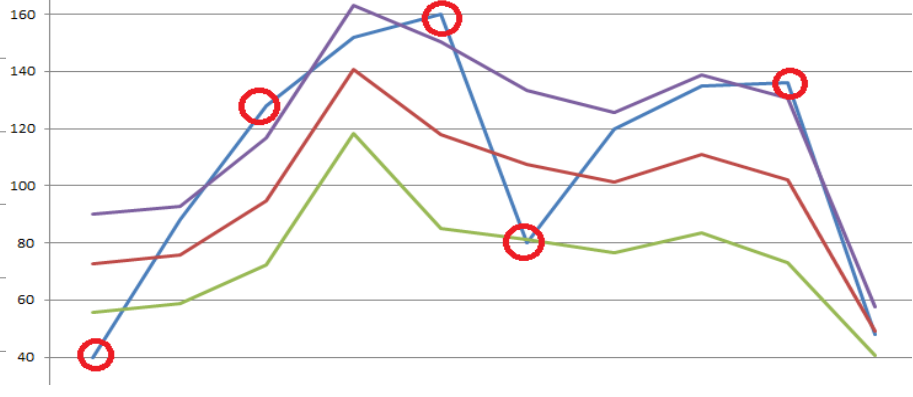
Feature	Uniqueness Score
How you mouse	.1 (1 out of 10 mouse like you)
The way you mouse	.1 (1 out of 10 mouse the same way)
How you type	.1 (1 out of 10 type like you)
The way you type	.1 (1 out of 10 type the same way)
Device, Application, Time of Day, Etc...	Etc...
Total	.1 x .1 x .1 x * .1 = 0.0__001 (potentially 1 in 10, __000 have the same cyber DNA signature)

User Signatures



Match

More closely follows average
Fewer measurements outside SD (21%)

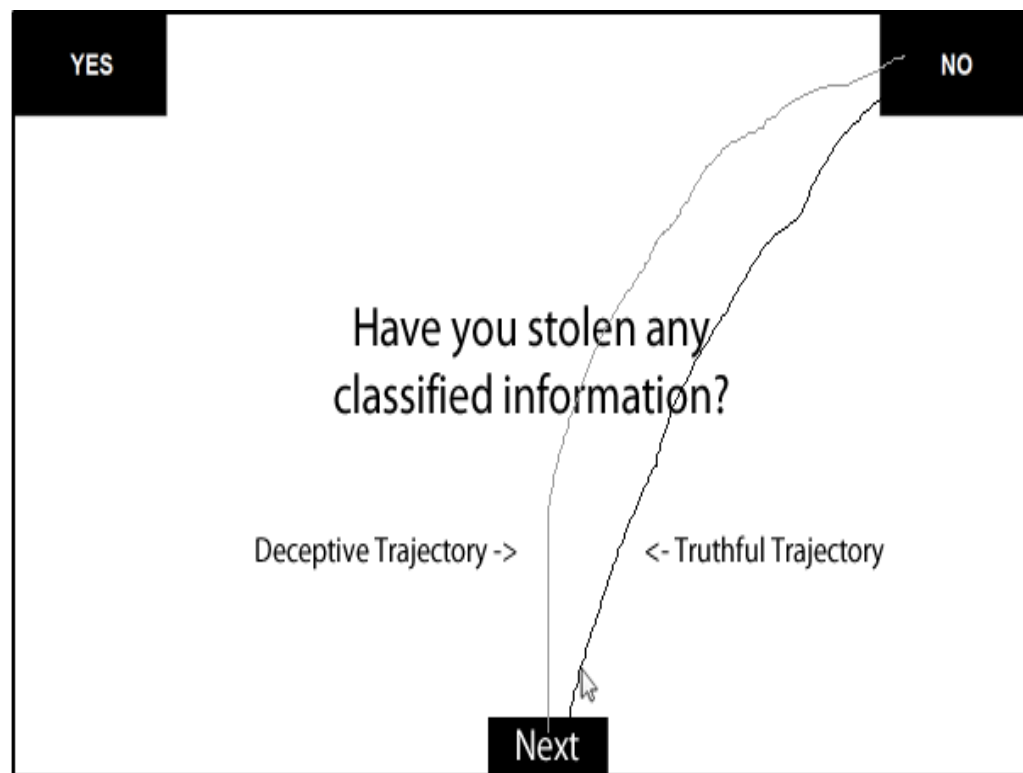


Intruder

More deviations from average
More measurements outside SD (50%)

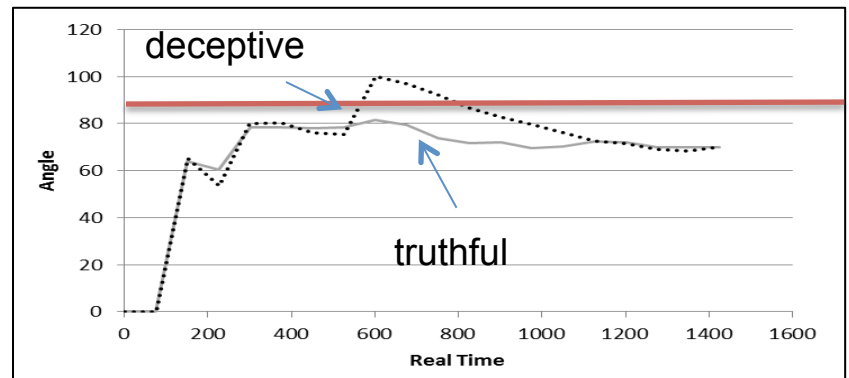
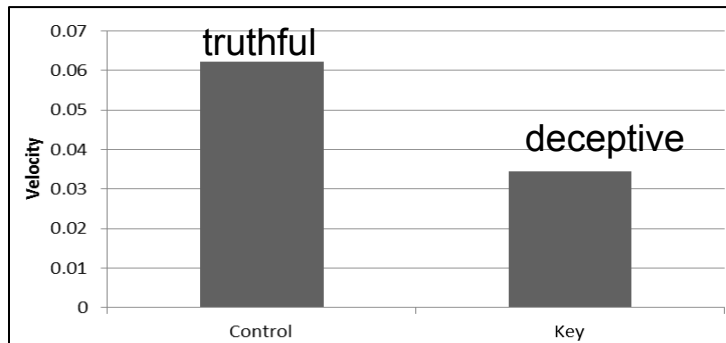
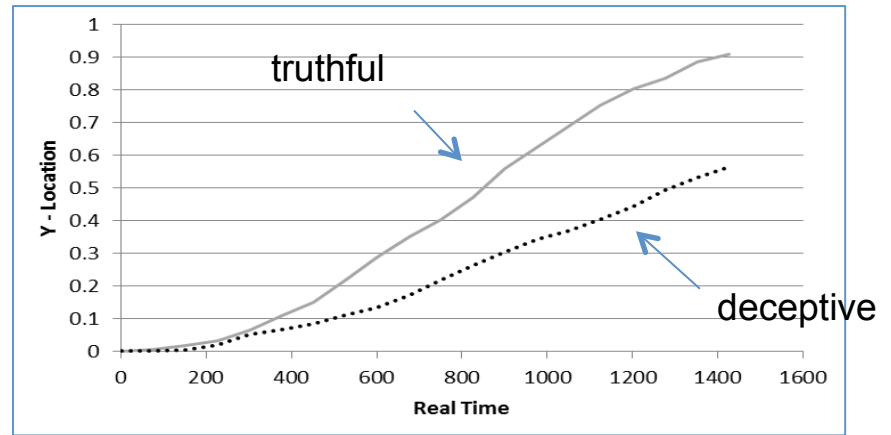
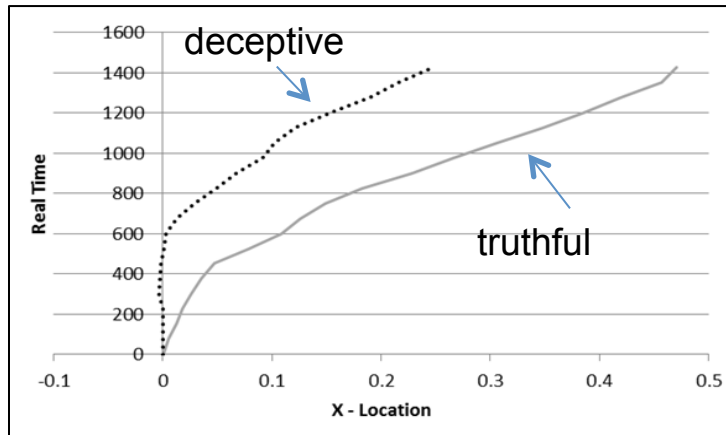
ADMIT: Automated Detection Method for Insider Threat

- Diagnose insider threats in screening surveys through monitoring mousing behavior
- Insider threats will likely show a difference in mouse movements for three reasons:
 - Cognitive conflict
 - Arousal
 - Task-Induced Search Bias



Selected Results

Within: Key questions (dotted lines) vs. Control questions (solid line) for insider threats



ADMIT Status

- 2000 human subjects tested
- Equal or beating polygraph
- Prototype built, Amazon cloud
- Building management dashboard, email deployment, and data visualization
- Looking for field test site

Application of ADMIT

- Insider Threat Event Investigation
- Employment screening
- Health care evaluations
- Annual (routine) employee integrity screening
- Life and other insurance applications
- Loan applications
- Testing
- Etc...

Project Novelty

- Mass deployable and easy to scale (Google of Polygraph)
- Language agnostic
- Bias free
- Unobtrusive
- Not easily fooled
- Guide which employees are not insider threats
- Discover networks of insider threats
- Many applications

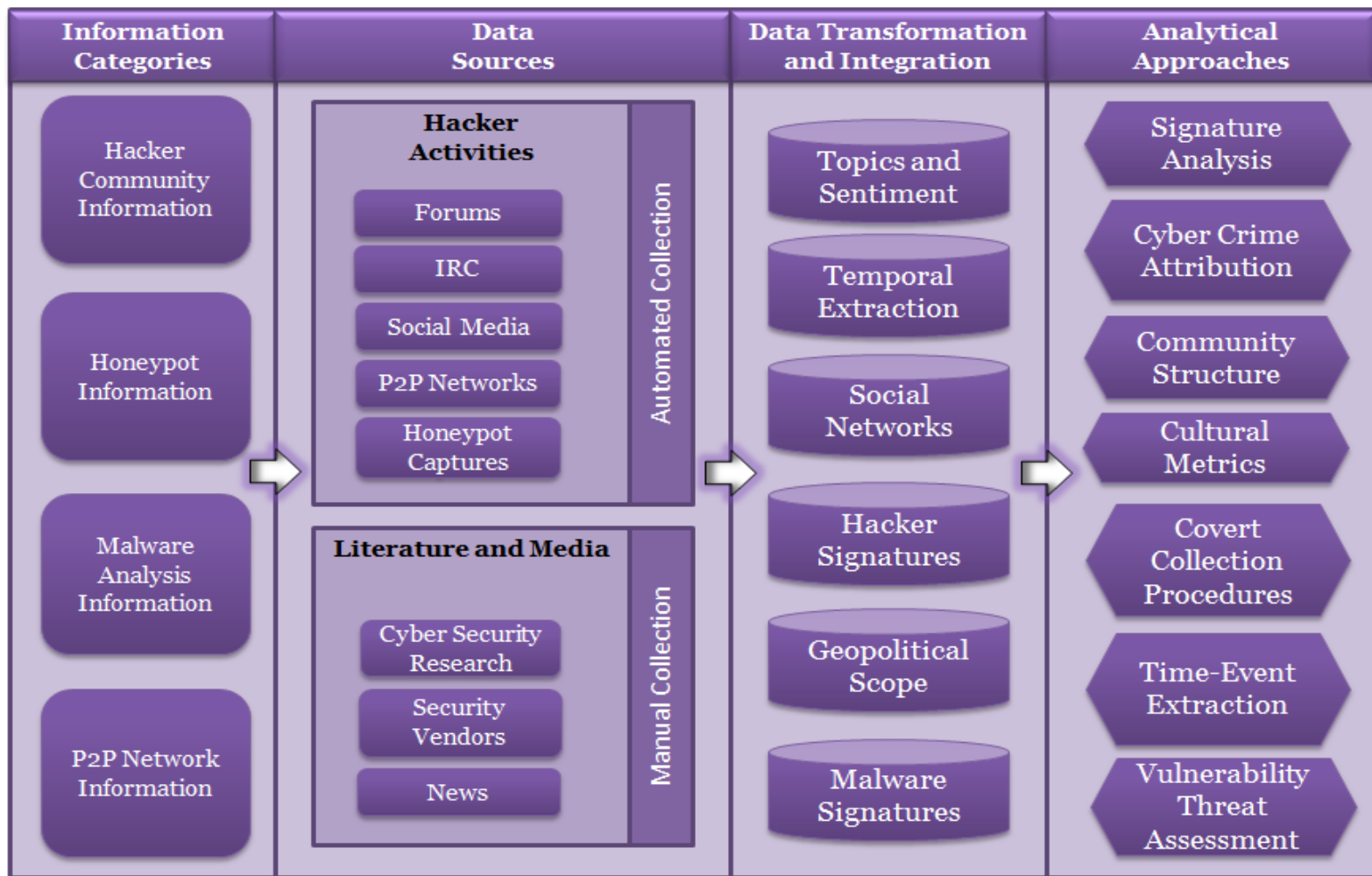
HACKER WEB



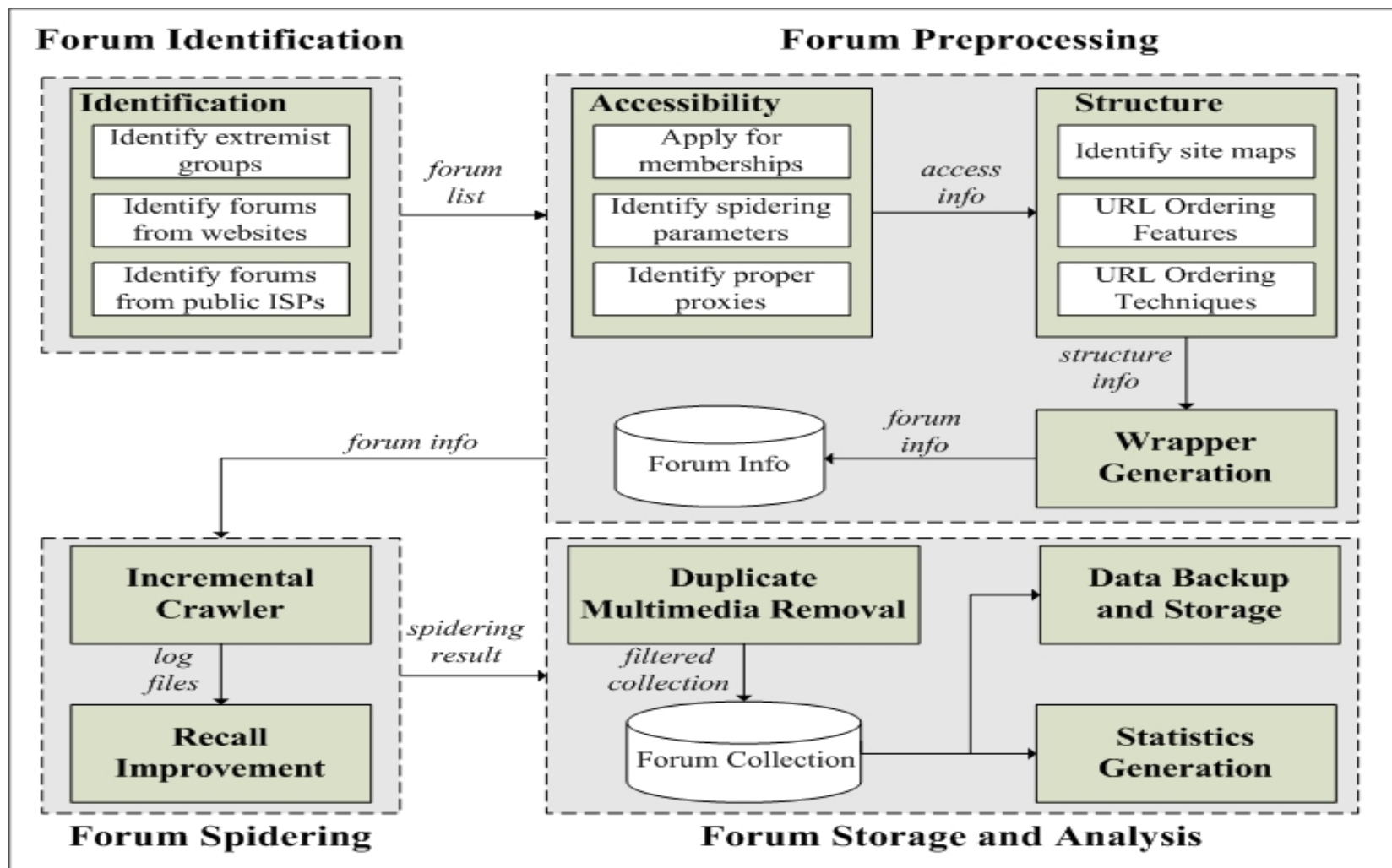
First Franco-American Workshop
October 17-18, 2013, Lyon France



Research Framework



Hacker Forum Collection

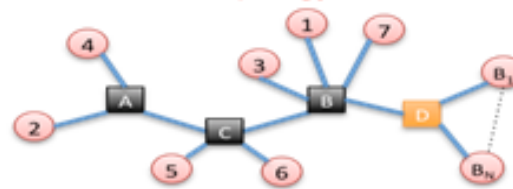


IRC Collection & Analytics

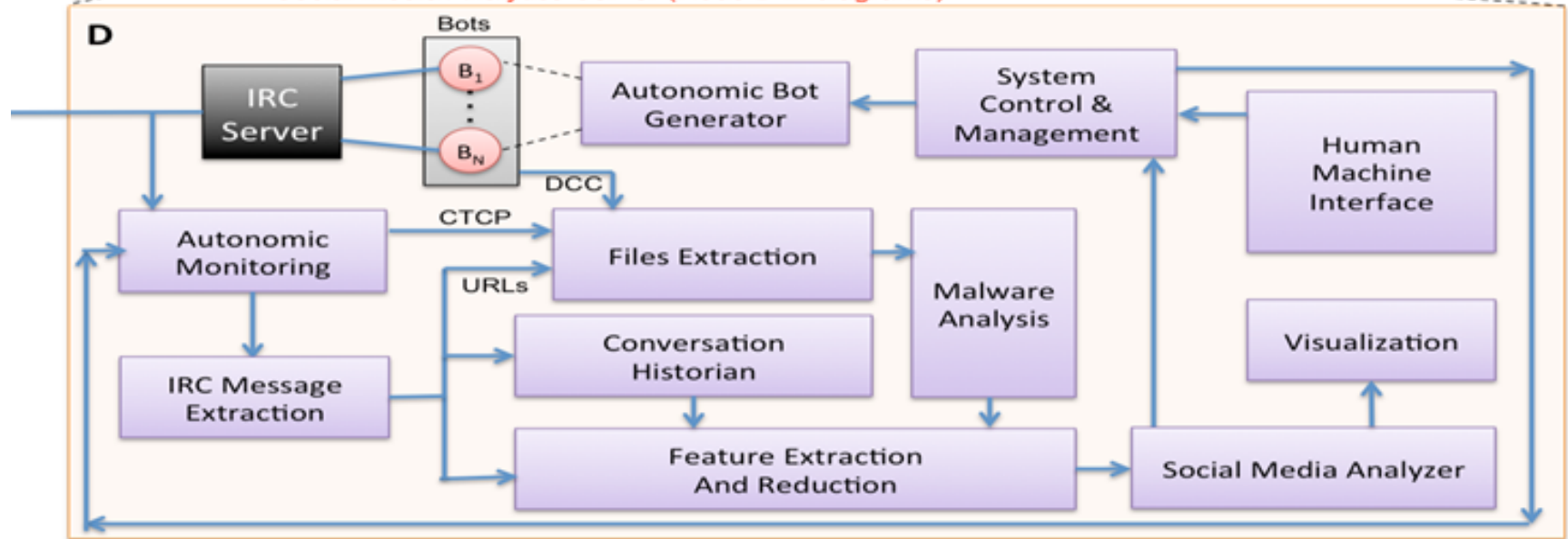
IRC Network



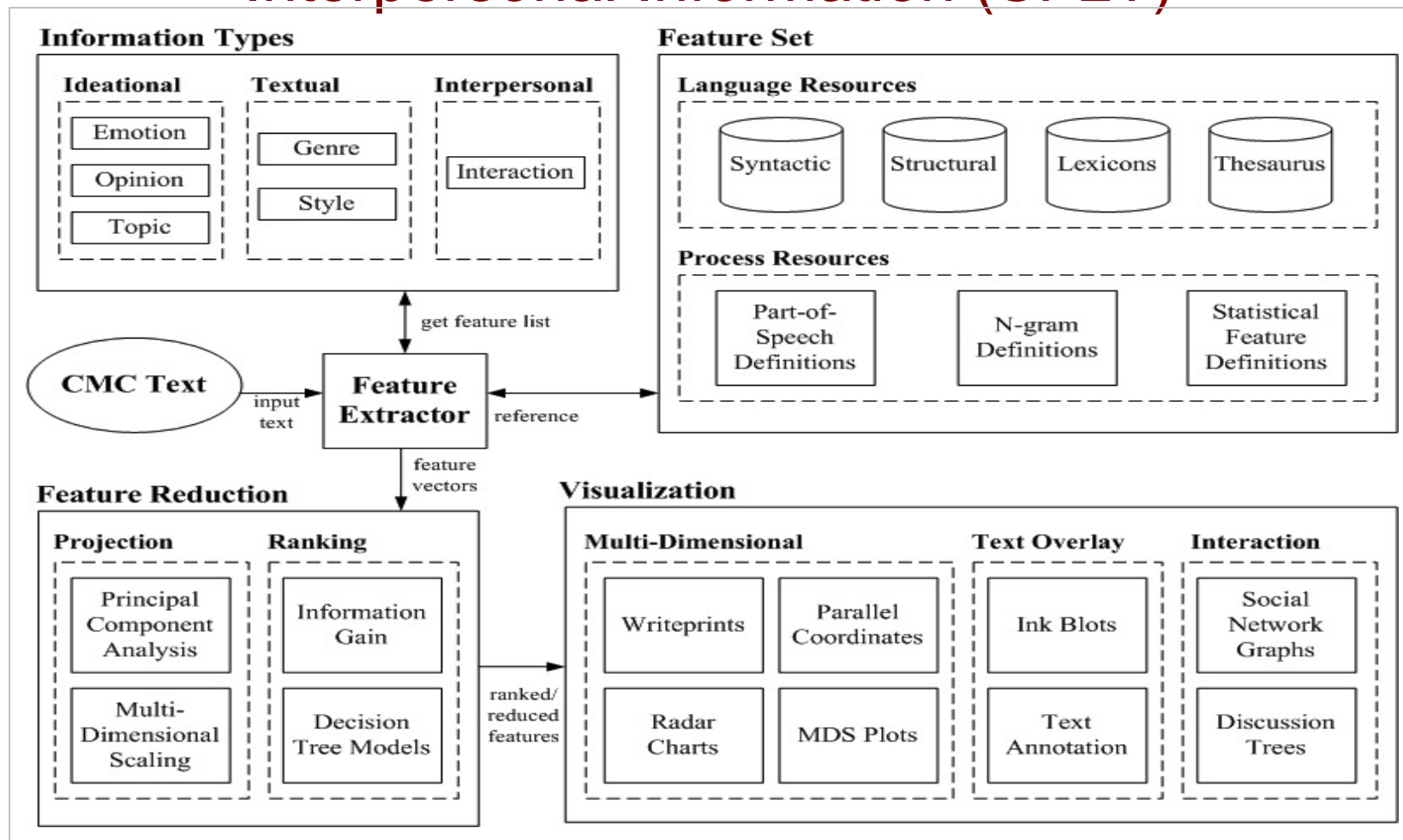
IRC Network Topology







Social Media Analytics Server (Node D in diagrams)





Social Media Analytics: Ideational, Textual and Interpersonal Information (SFLT)



Project Objectives

-  Develop autonomic monitoring and analysis of IRC hacker messages
-  Build an IRC testbed, to experiment with and evaluate the effectiveness of our tools and algorithms
-  Identifying hackers relations, behaviors, and interactions.
-  Identifying IRC based botnet



IRC Background

-  The IRC (Internet Relay Chat) protocol is a text based conferencing protocol, which has been developed in 1989.
-  The IRC protocol is based on the client/server model, and it is designed to run in a distributed manner.

IRC Background


- 🐾 The simplest architecture consists of a server with multiple clients connect to it.
- 🐾 The server will handle message delivery and multiplexing.
- 🐾 There are two types of clients:
 - user clients
 - service clients.

IRC Background


-  The user clients are text-based interfaces that interactively communicate using IRC.
-  The service clients are used to provide services to user clients, such as providing statistics.


IRC Background

 *Servers relay all the communications between the clients.*

 *All messages from any server are broadcast to all the other connected servers.*

Proposed Techniques

-  1) IRC Server based technique

-  2) IRC client based technique

IRC Server based technique

The environment from outside will look like a regular IRC server that will be registered with one of the well-known IRC networks. The environment will consist of the following components:

- 🐱 **IRC Server:** This is a regular IRC server, which will allow the interaction with the rest of the IRC network and also logging IRC messages, since all IRC servers receive information from all the nodes in the network.
- 🐱 **Autonomic Monitoring:** This component is responsible for picking up all the IRC packets, and it will have policies that define which ports to monitor and when.



IRC Server based technique

- 🐱 **IRC Message Extraction:** This component will extract IRC messages from IRC packets, and categorize that into different IRC message types.
- 🐱 **File Extraction:** It will be responsible for detecting file transfer and extracting files from communications. This will work with DCC transfers or URLs.
- 🐱 **Conversation Historian:** This module will be responsible for building conversations from the IRC messages and storing those for analysis.
- 🐱 **Malware Analysis:** This module will use different tools to detect if the shared files contain malware (Viruses, worms, Trojans, ...).
- 🐱 **Feature Extraction and Reduction:** This module will extract all the features needed to perform the analysis from the IRC Messages, Conversation History, and the Results of the Malware analysis. It will also reduce the complexity of the extracted features.

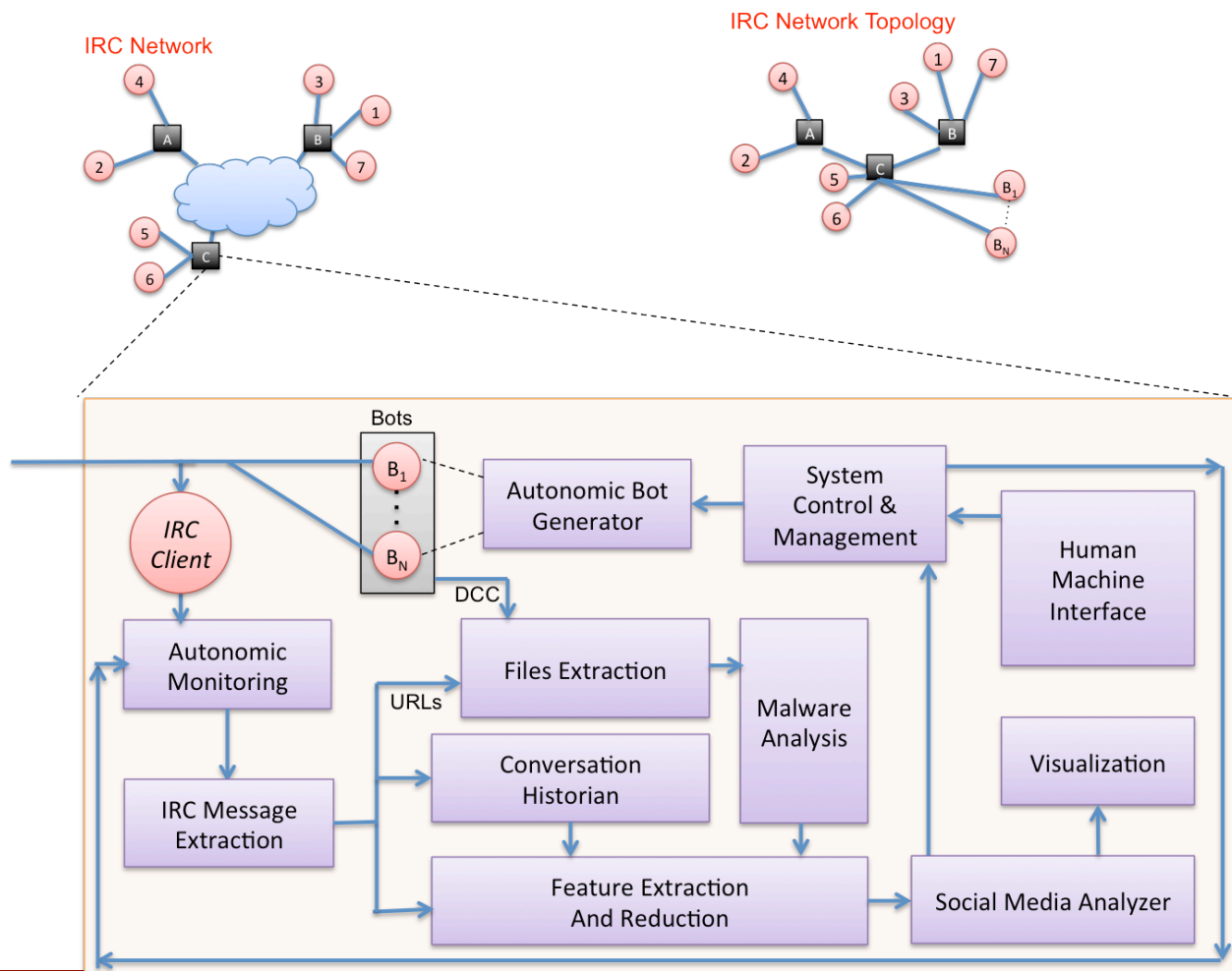
IRC Server based technique

- 🐱 **Social Media Analyzer:** This is the core of the system, and it will be responsible for detecting, classifying, measuring, and tracking the formation, development, and spread of topics, ideas, and concepts in cyber attacker social media communication. It will also identify important and influential cyber criminals and their interests, intent, sentiment, and opinions in online discourses. And it will induce and recognize attacker identities, online profiles/styles, communication genres, and interaction patterns.
- 🐱 **Visualization:** This module will provide an insight of strategic communication in critical social media.
- 🐱 **Autonomic Bot Generator:** This component is responsible for generating Bots that provide interaction mechanism with the environment. The bot behaviors, types, and number is enforced based on a preset policy.

IRC Server based technique



-  **Human Machine Interaction (HMI):** This provides the interface for administrator to control the environment.
-  **System Control and Management:** This module is the one responsible for setting the policies based on the current environment situation and/or according to the administrator command through the HMI.

IRC Client based technique



IRC Client based technique

The environment will consist of the following components:

-  **IRC Clients and Bots:** These are regular IRC clients and Bots, which will allow the interaction with the rest of the IRC network and also logging IRC messages.
-  **Autonomic Monitoring:** This component is responsible for picking up all the IRC packets, and it will have policies that define which ports to monitor and when.



IRC Client based technique

- 🐱 **IRC Message Extraction:** This component will extract IRC messages from IRC packets, and categorize that into different IRC message types.
- 🐱 **File Extraction:** It will be responsible for detecting file transfer and extracting files from communications. This will work with DCC transfers or URLs.
- 🐱 **Conversation Historian:** This module will be responsible for building conversations from the IRC messages and storing those for analysis.
- 🐱 **Malware Analysis:** This module will use different tools to detect if the shared files contain malware (Viruses, worms, Trojans, ...).
- 🐱 **Feature Extraction and Reduction:** This module will extract all the features needed to perform the analysis from the IRC Messages, Conversation History, and the Results of the Malware analysis. It will also reduce the complexity of the extracted features.

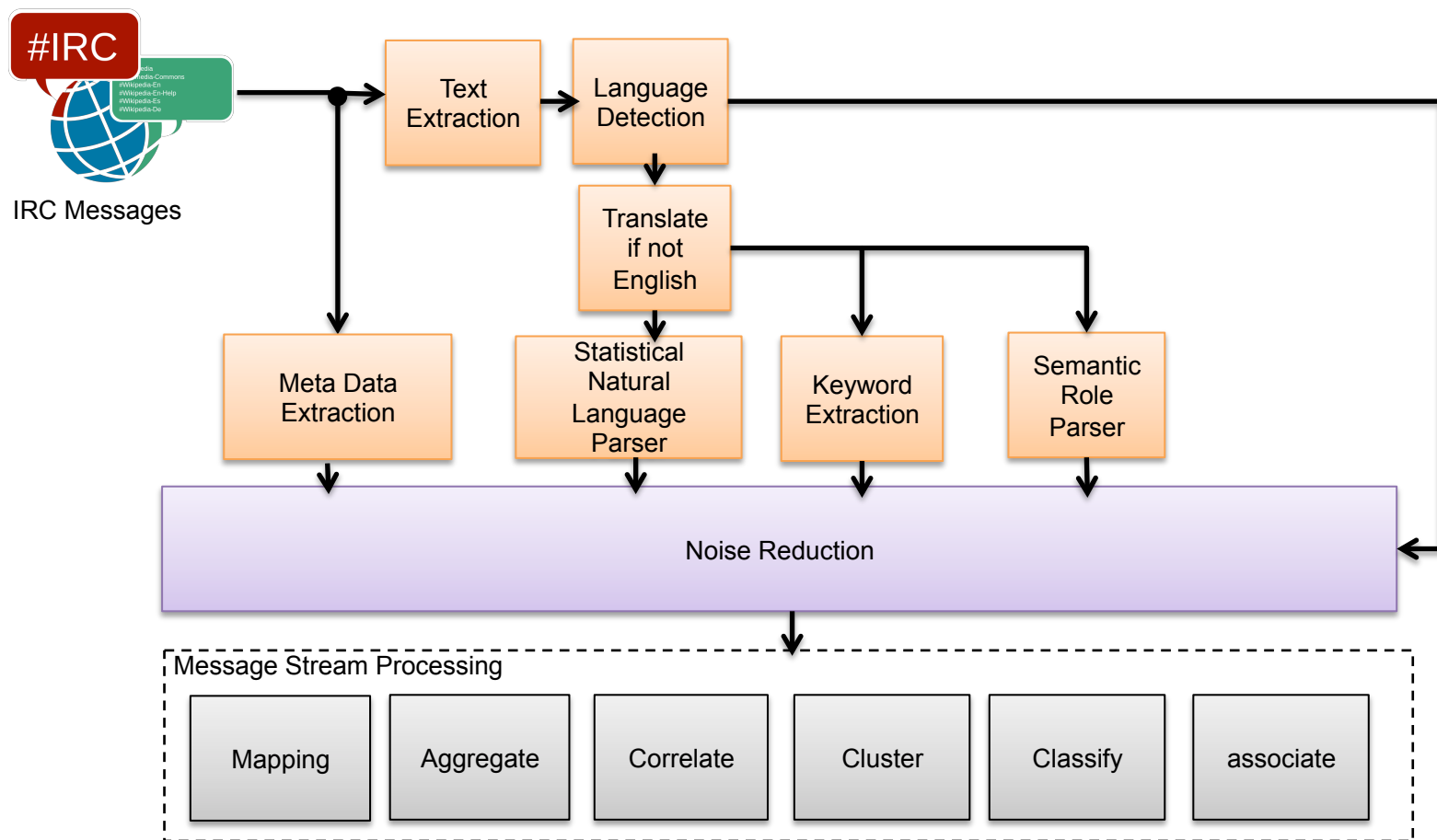
IRC Client based technique

- 🐱 **Social Media Analyzer:** This is the core of the system, and it will be responsible for detecting, classifying, measuring, and tracking the formation, development, and spread of topics, ideas, and concepts in cyber attacker social media communication. It will also identify important and influential cyber criminals and their interests, intent, sentiment, and opinions in online discourses. And it will induce and recognize attacker identities, online profiles/styles, communication genres, and interaction patterns.
- 🐱 **Visualization:** This module will provide an insight of strategic communication in critical social media.
- 🐱 **Autonomic Bot Generator:** This component is responsible for generating Bots that provide interaction mechanism with the environment. The bot behaviors, types, and number is enforced based on a preset policy.





IRC Client based technique

-  **Human Machine Interaction (HMI):** This provides the interface for administrator to control the environment.
-  **System Control and Management:** This module is the one responsible for setting the policies based on the current environment situation and/or according to the administrator command through the HMI.

Features Extraction and Reduction from IRC Messages



Conclusions

-  We cannot build perfect network centric systems for the next generation Internet Of Everything's (IoE) services
-  Autonomic computing provides a promising paradigm to self manage and self-protect next generation IoE services
-  Resilient techniques based on Software Behavior Encryption and Moving Target Defense can lead to the development on Intrusion Tolerance Systems (ITS)
-  Anomaly behavior analysis will help driving when to change the environment and respond optimally to attacks or malicious events.

THANK YOU



First Franco-American Workshop
October 17-18, 2013, Lyon France

