Developing a Requirements Framework for Cybercraft Trust Evaluation

J. Todd McDonald
Shannon Hunt

Center for Cyberspace Research
Department of Electrical and Computer Engineering
Air Force Institute of Technology
Wright Patterson AFB, OH

Sponsor

Research sponsorship by:

Cybercraft Initiative

AFRL/RIGA

Cyber-Operations Branch

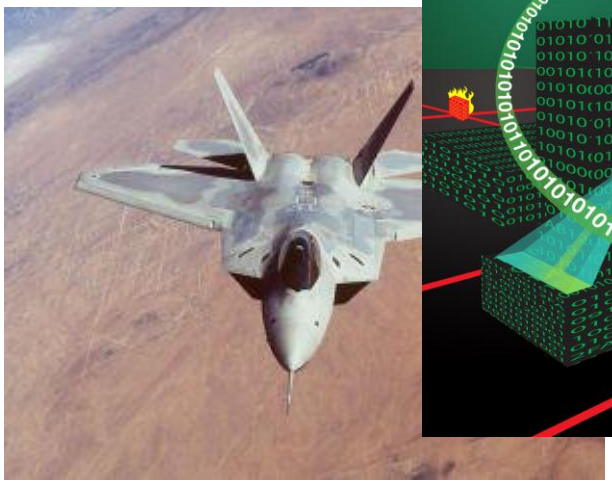
Rome Labs, NY

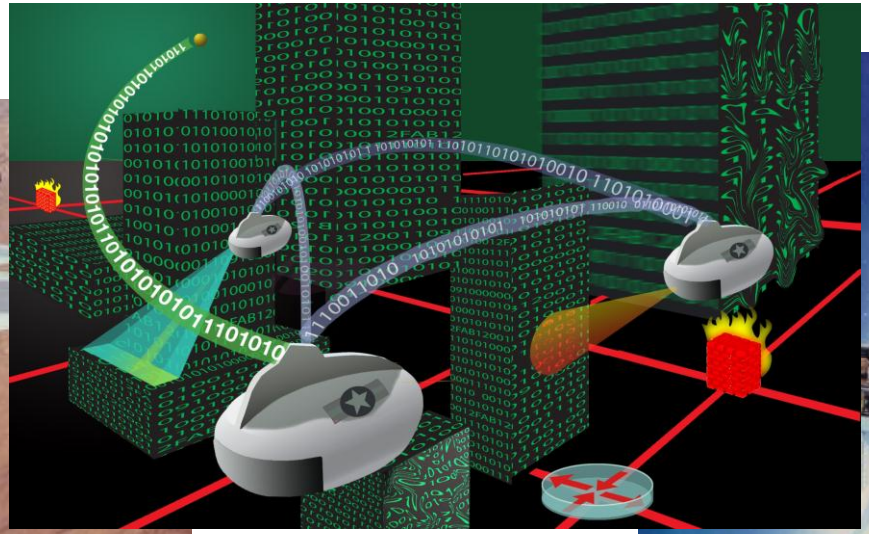
Context: Air Force Mission

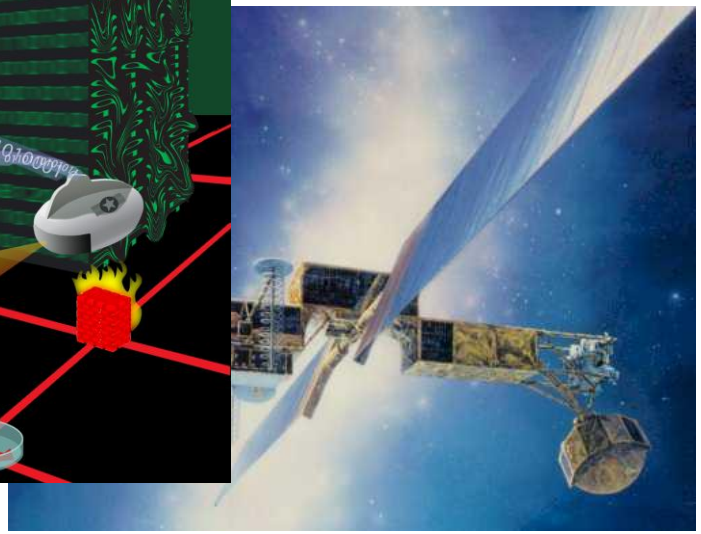
"*The mission of the United States Air Force is to deliver sovereign options for the defense of the United States of America and its global interests -- to fly and fight in Air, Space, and* ***Cyberspace****.*"

- Michael W. Wynne

Cybercraft: *Cyberspace Superiority*







Aircraft: *Air Superiority*

Spacecraft: *Space Superiority*

What is a Cybercraft?

"A Cybercraft is a trusted computer entity designed to cooperate with other Cybercraft to defend Air Force networks."

- Cybercraft fleet
  - Composed of autonomous agents
  - Installed on every AF network device (1+ million agents)
  - Incorporate decision engines to rapidly make decisions and take defensive actions without human intervention
  - Command and Control network to pass commands, policies, environment data, payloads, etc.

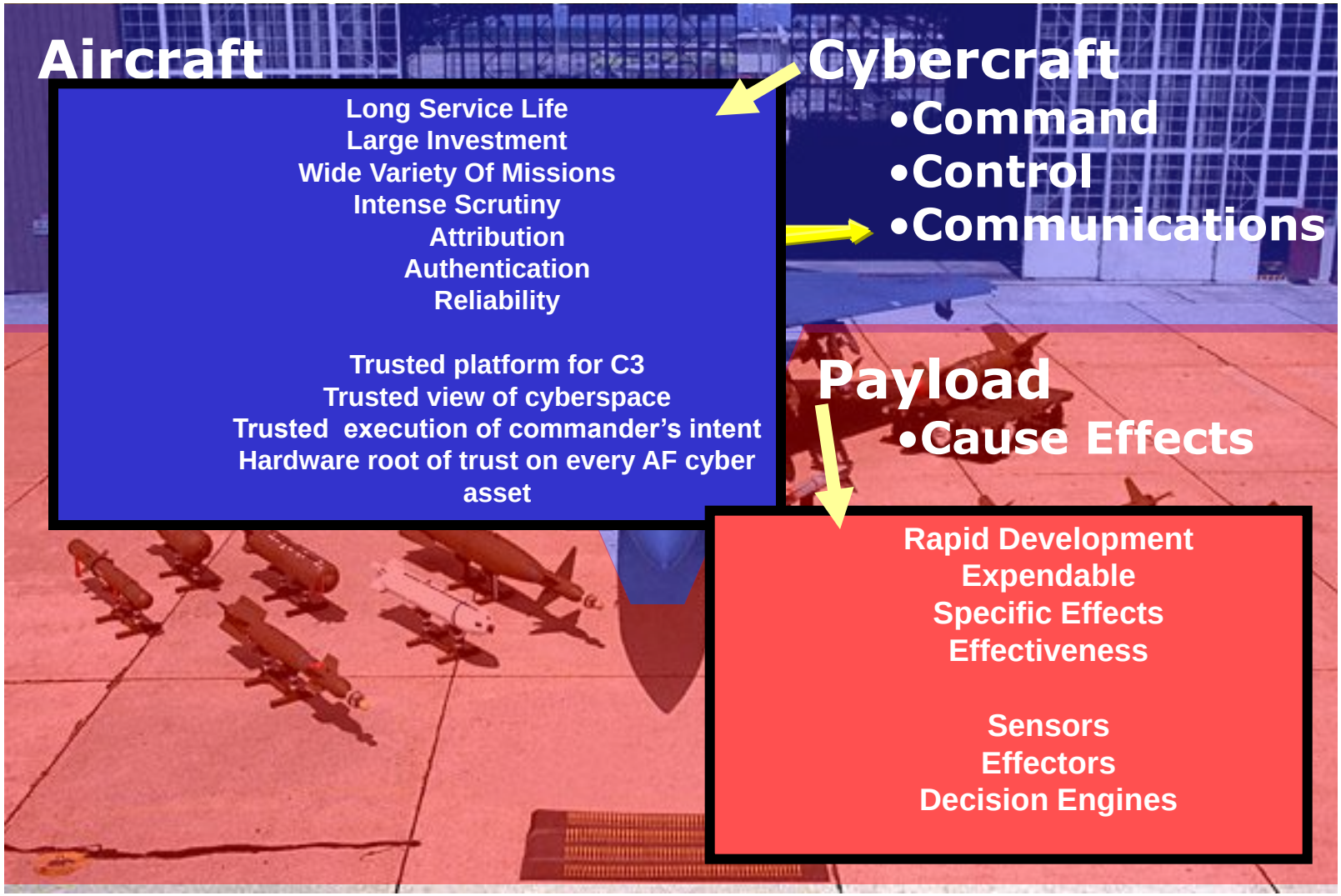
What is required for a commander to *trust* a Cybercraft to act autonomously to defend military information systems?

Motivation & Goals

- Can we create a reference framework for evaluating various trust models and their applicability for use in Cybercraft?
  - Can we link specific Cybercraft scenarios to specific trust model expressions?
  - Can we express and evaluate transitive trust for specific Cybercraft mission scenarios?

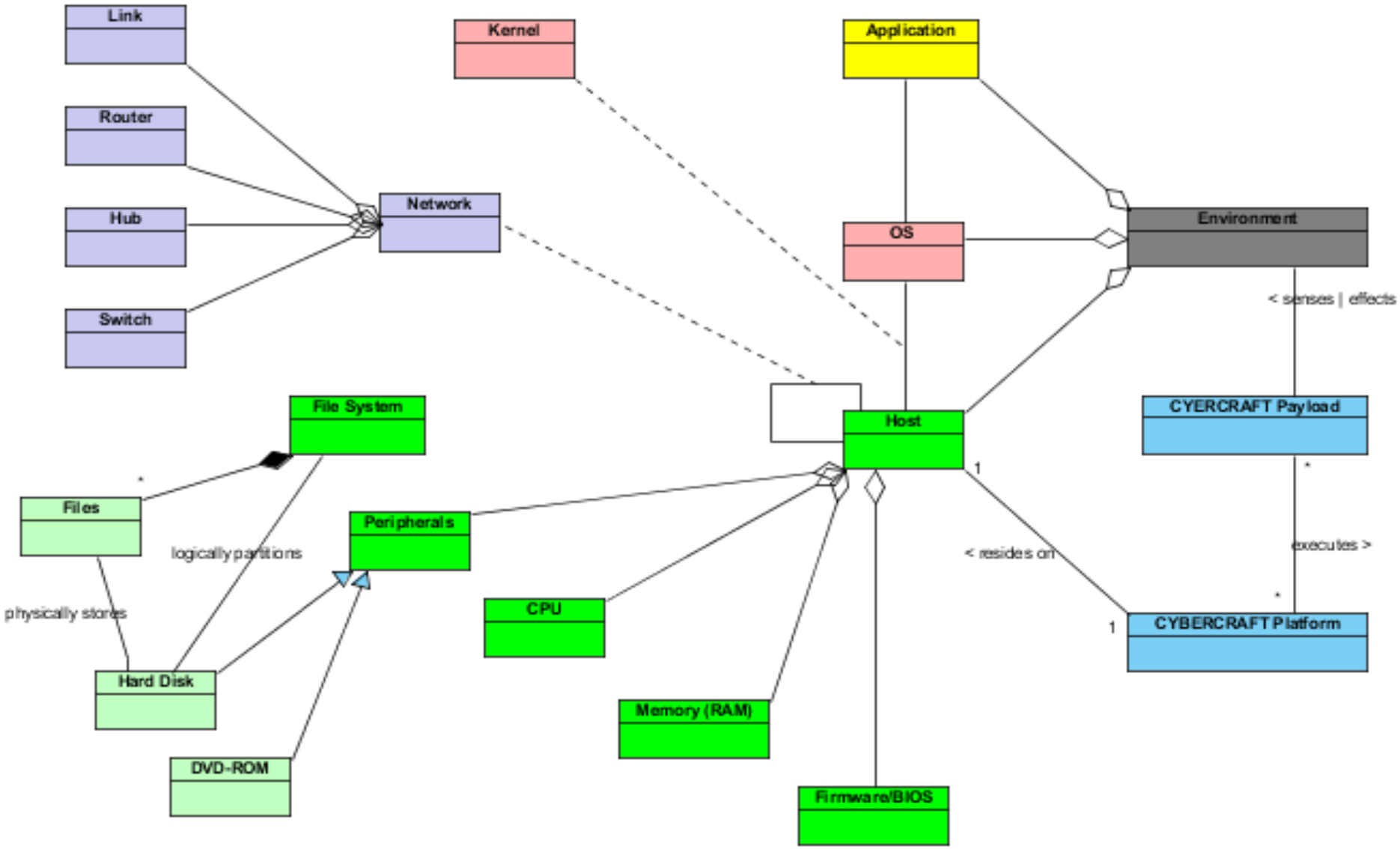
This research presents an approach for considering trust expression in relation to Cybercraft requirements, analysis, and design consideration

Conceptual Architecture



Cybercraft Domain

Develop America's Airmen Today ... for Tomorrow



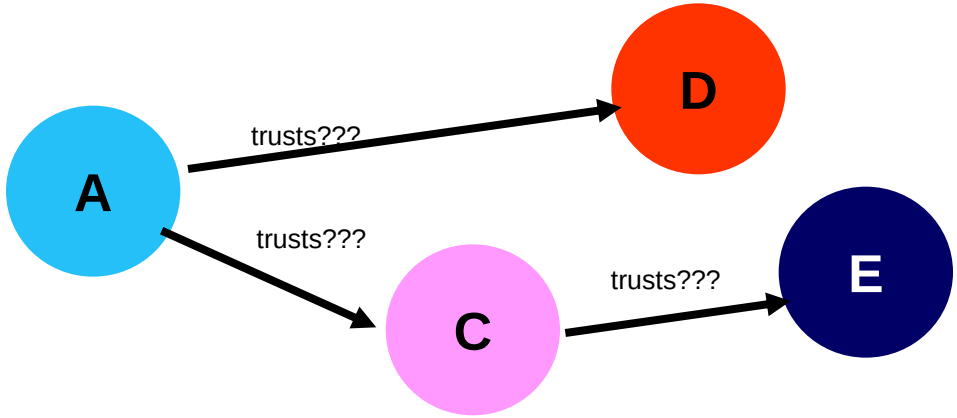
Trust in Cybercraft

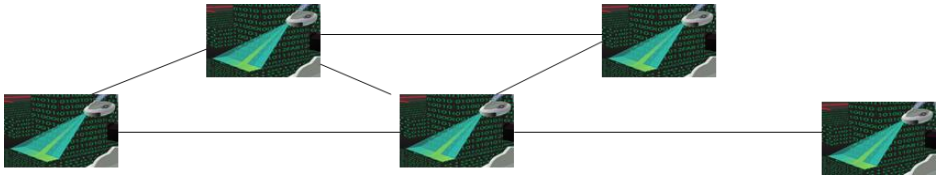
- Why bother with trust (yuck, it's elusive) versus security anyway ???
  - Non-human autonomy / decision making
  - Ability to characterize human-like decision making process
- Root of trust (platform)
  - Hardware versus software protection (virtualization/OS level)
  - Transitivity from platform to payloads
- Trust in an agent's abilities (platform/payload)
  - Confidence in the data produced by an agent
  - Identify which agents may be compromised or are incompetent
- Limitation of powers (payload)
  - Policy-defined bounds for autonomous decisions
  - How not to create a DDOS threat from our own Cybercraft fleet
  - Establishing commander-level trust in boundaries
- Depiction of the environment (payload)
  - Combining data produced by different agents
  - Estimating the effectiveness of a Cyber-operation (Cyber BDA)

Transitive Trust

- A → B → C → D → E
  - Read A trusts B, who trusts C, who trusts D, who trusts E, therefore A trusts E

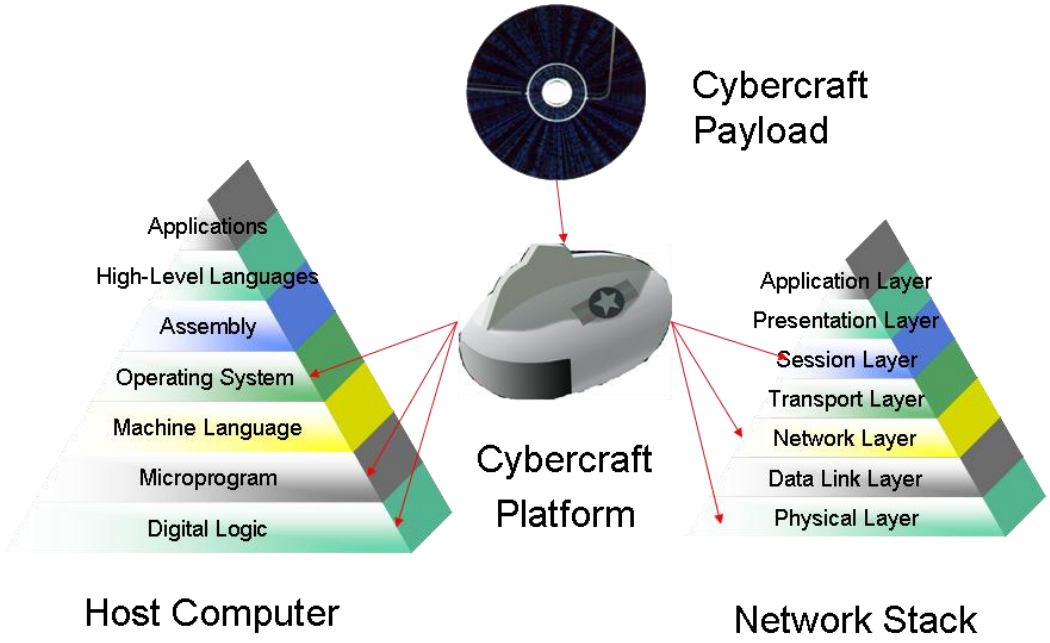
- Possibilities assessments
  - Platform to platform
  - Agent to agent (payloads)
  - Platform to agent (payload)
  - Platform to environment
  - Payload to environment





Root of Trust

- Does the root of trust in the Cybercraft platform transfer to the other components of the system
  - OS
  - Network
  - Applications
  - Third-party software



Software Process Models vs. Trust Models

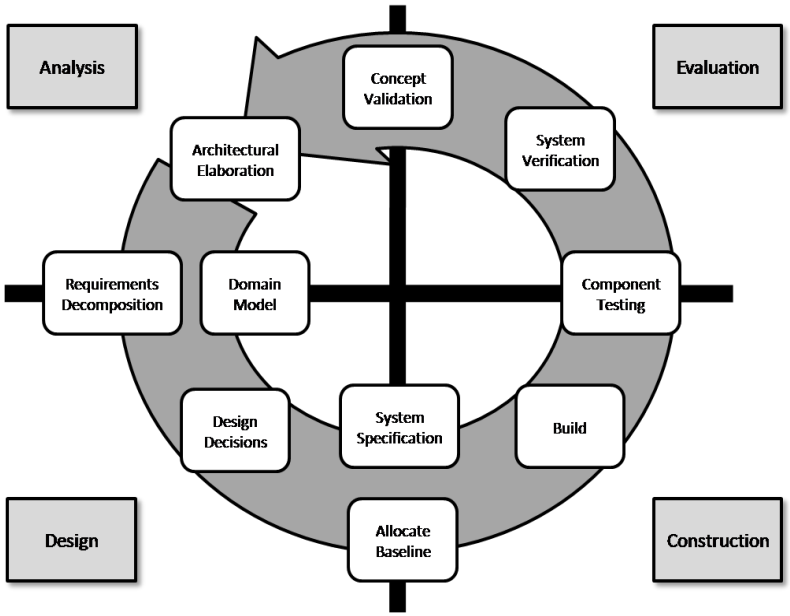
Software Process Models

- Specification-based (waterfall)
  - Usage of prototyping
- Iterative / Evolutionary processes
  - Incremental delivery
  - Spiral development
  - Agile development
  - Rational Unified Process
  - Extreme Programming

Trust Models

- Allows for a mathematically way to gauge trustworthiness of interacting entities
  - Enable devices to form, maintain, and evolve trust opinions
    - Opinions are used for the configuration of the system
  - Incorporate Quality of Service (QoS) requirements
    - Whether or not certain transactions with take place or not (low – high risk)
  - Plan for the lack of a globally available infrastructure
  - Entities that are dynamic and anonymous
  - Human tailored
  - Subjective
  - Highly customizable

Bridging Trust and Requirements



- How do we transition from user requirements to evaluating commander's trust?
- How do we express agent-based trust in terms of system usage and possible mission areas?

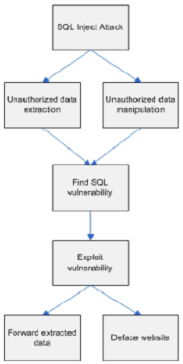
- We need models to precisely evaluate security assumptions, attacks, and risks within the Cybercraft architecture
- We need a mathematical approach to understanding transitive trust and root of trust questions specific to Cybercraft missions

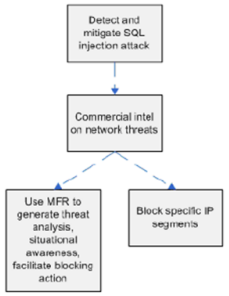
"It is essential that regardless of the (trust) model chosen, the reason we want to use the model and our expectation of what it will provide in terms of security must be clearly defined."

Requirements Analysis

- **Explicit Cybercraft requirements are immature, therefore explicit *trust* model requirements are immature**

- Solution: Provide iterative approach
  - Attack/Defense Trees
    - Visualize attacks on our networks and ways to defend them
  - Use Cases
    - Text describing step-by-step interaction between a user and a system

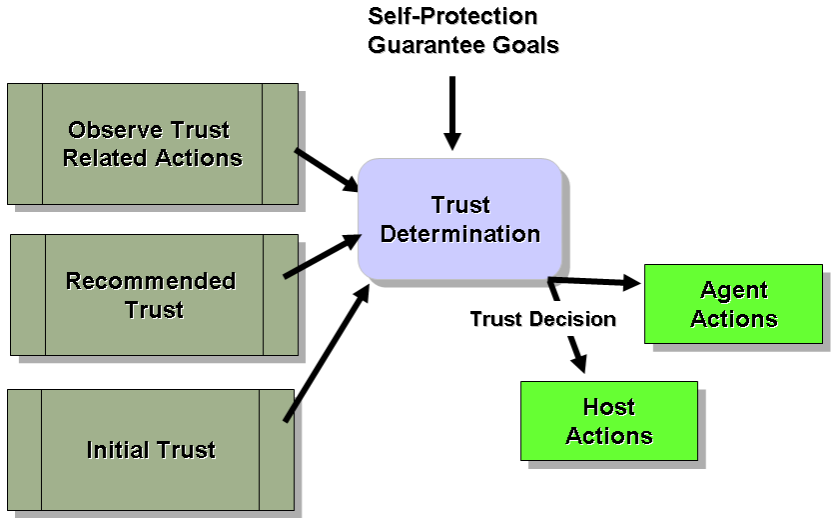




Use Case Name	Anti-Virus
Scope	The network
Level	Ensure anti-virus software is installed and up-to-date on all machines
Primary Actor	Cybercraft
Stakeholders and Interests	Network Defenders
Preconditions	Network is operational, up-to-date,
Success Guarantee	All machines have anti-virus software loaded, operational, and up-to-date
Main Success Scenario	Cybercraft platform creates a payload to check anti-virus software on all machines in the network, if all machines have operational AV that is up-to-date, the scenario is successful
Extensions	Cybercraft platform creates a payload to check anti-virus software on all machines in the network. Alternate scenarios: 1. If there is no AV software, the Cybercraft platform dispatches another payload to install AV software on the machine in question 2. If there is AV software installed, but not updated, the Cybercraft platform dispatches another payload to obtain correct updates from approved sites
Frequency of Occurrence	Daily
Miscellaneous	Assumptions are that the Cybercraft payload and platforms are trusted, the network is secure, all channels a Cybercraft uses are secure

Trust Model Evaluation

- Three main ideas of trust
  - initial trust
  - trust exchange
  - trust evolution
- Three models under view
  - hTrust (human Trust)
  - VTrust (Trust Vector)
  - P2P (Peer to Peer)
- Applying the models:
  - Evaluate fitness of models for Cybercraft trust questions
  - Apply specific scenarios



Trust Model	Initial Trust	Trust Exchange	Trust Evolution
hTrust	formation	dissemination	evolution
VTRUST	knowledge	experience	recommended
P2P	ratings generation	ratings discovery	ratings aggregation

Current Scenarios

- Scenario One – transitive trust
  - How far can each model create a transitive trust chain (a → b → c → d → e …)
- Scenario Two – AV update
  - Case one: AV is installed on machine and up-to-date
  - Case two: AV is not installed
  - Case three: AV is installed but not updated

Agent	Value
A	Cybercraft platform
B	Cybercraft payload check
C	Cybercraft payload update
D	Cybercraft payload install
E	OS
F	Network
G	AV software on OS (agent E)
H	Update place
I	AV software from network

Scenario 1 Analysis

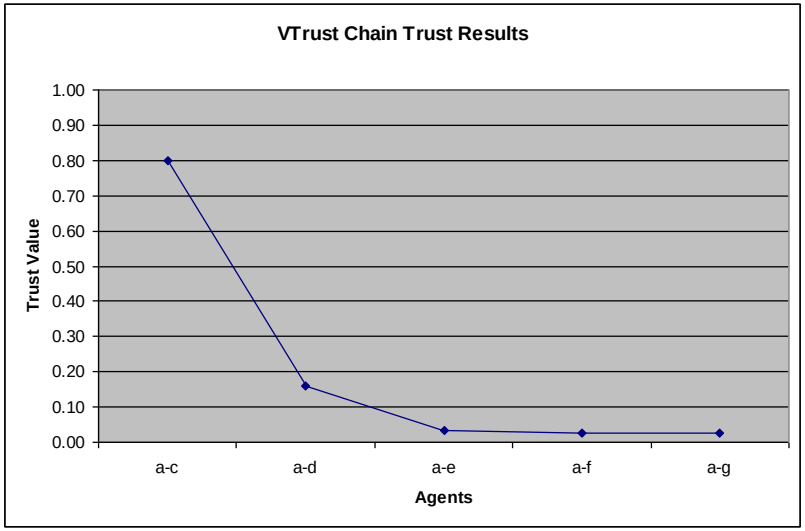
- hTrust – chain fell apart after agent *c*
- P2P – chain can be quite long
- VTrust – depends on the values

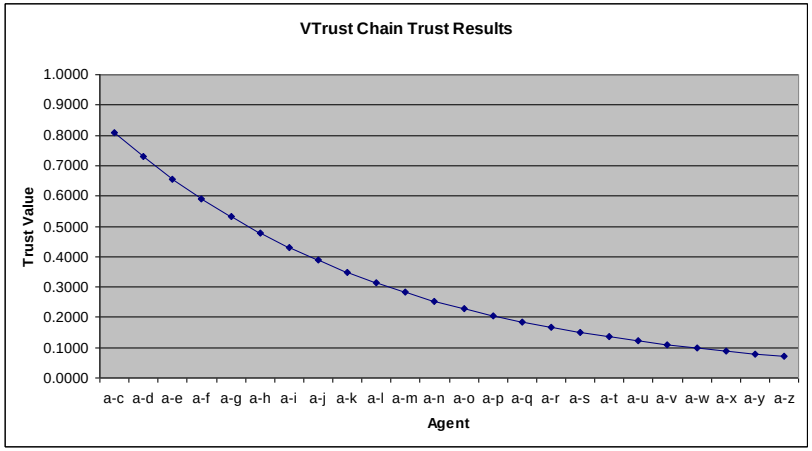
VTrust initial values

Trustor Initial Recommendation Values	
$(A \rightarrow B)_t^N$	1.0
$(B \rightarrow C)_t^N$	0.8
$(C \rightarrow D)_t^N$	0.2
$(D \rightarrow E)_t^N$	0.2
$(E \rightarrow F)_t^N$	0.8
$(F \rightarrow G)_t^N$	1.0

VTrust final results

Recommendation Chain Results	
$A \rightarrow C$	0.80
$A \rightarrow D$	0.16
$A \rightarrow E$	0.032
$A \rightarrow F$	0.0256
$A \rightarrow G$	0.0256

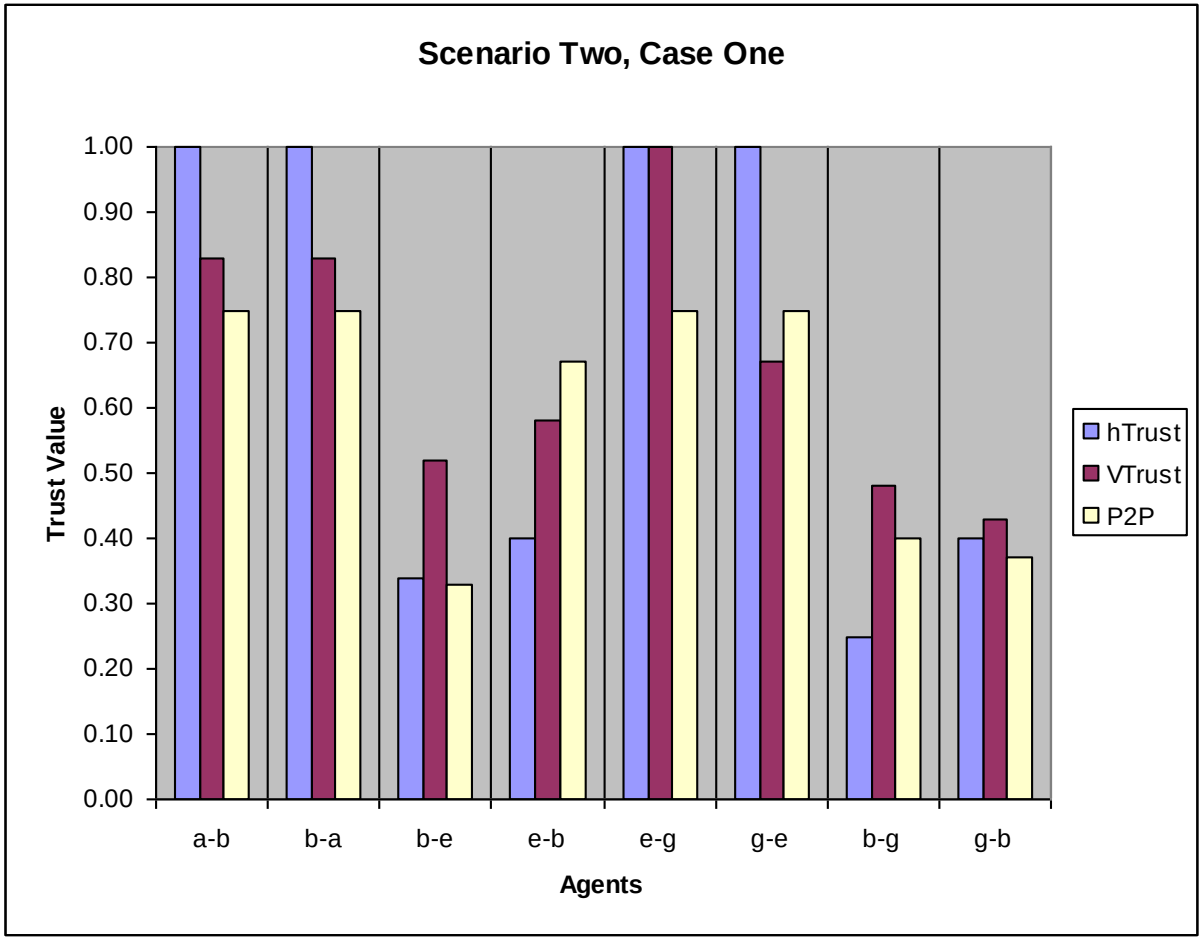




Scenario 2 Analysis

Case One: AV is installed on machine and up-to-date

A, B, E, G



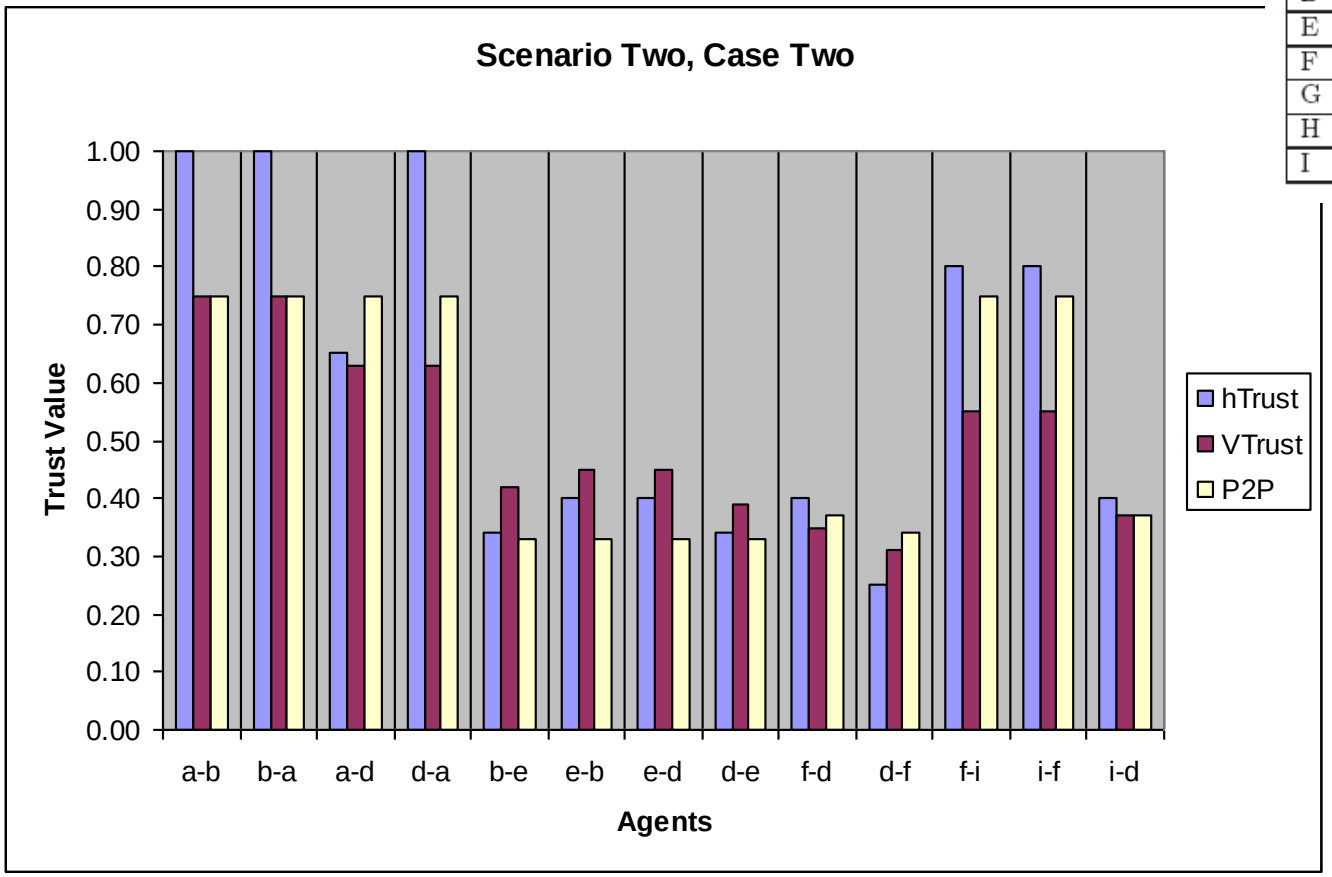
Agent	Value
A	Cybercraft platform
B	Cybercraft payload check
C	Cybercraft payload update
D	Cybercraft payload install
E	OS
F	Network
G	AV software on OS (agent E)
H	Update place
I	AV software from network

Scenario 2 Analysis

Case Two: AV is not installed

Agent	Value
A	Cybercraft platform
B	Cybercraft payload check
C	Cybercraft payload update
D	Cybercraft payload install
E	OS
F	Network
G	AV software on OS (agent E)
H	Update place
I	AV software from network

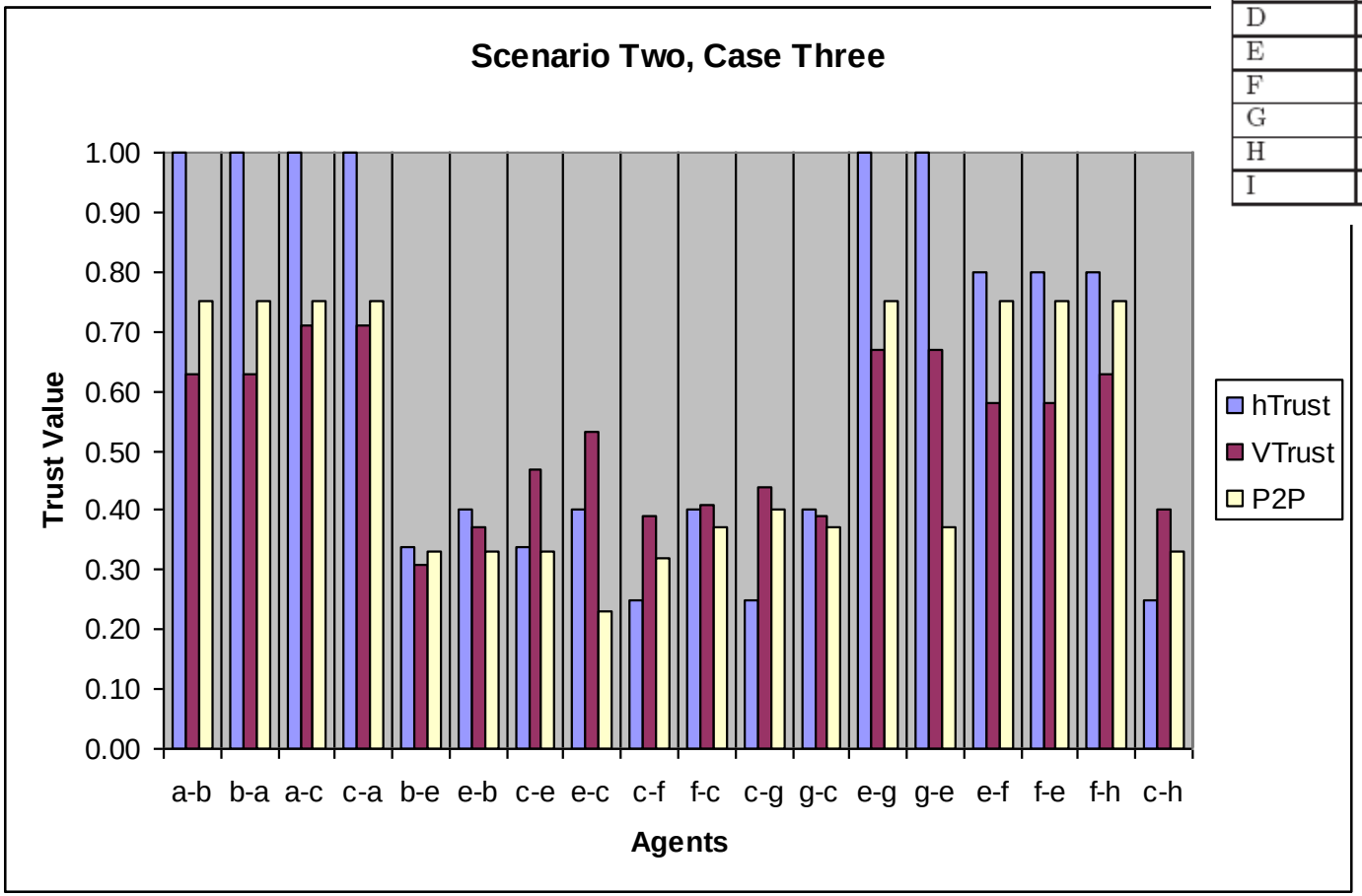
A, B, D, E, F, I



Scenario 2 Results

Case Three: AV is installed but not updated

A, B, C, E, F, G, H



Agent	Value
A	Cybercraft platform
B	Cybercraft payload check
C	Cybercraft payload update
D	Cybercraft payload install
E	OS
F	Network
G	AV software on OS (agent E)
H	Update place
I	AV software from network

Reference Framework

	hTrust	VTrust	P2P
Able to form, maintain, and evolve trust opinions	Yes	Yes	Yes
Incorporates QoS	Yes	Yes	Yes
Human tailored	Yes	No	No
Subjective	Yes	Yes	Yes
Highly customizable	Yes	No	Yes
Allows for transitive trust	No	Yes	Yes
Dynamic trust changing	Yes	Yes	Yes
Minimal resource demands	Yes	Yes	Yes

Some Contributions

- We provide a unique approach to requirements definition based on:
  - Use Case Analysis
  - Attack/Defense Trees
  - Mission Level Task Breakdown
- We provide specific correlation between abstract trust models and the Cybercraft trust problem related to specific system requirements
- We implement and analyze specific models to demonstrate the utility of trust expression within the context of Cybercraft
- We define a reference framework for evaluating existing and future trust models as well as provide specific measures for analyzing transitive trust relationships in view of the Cybercraft platform and its root of trust

Questions