FRAMEWORK FOR TRUSTWORTHY AUTONOMY

36th Soar Workshop
Ann Arbor, Michigan
Problem – Cyberspace Related Issues

Increasing Complexity

- Space Shuttle: ~400K LOC
- F22 Raptor fighter: ~2M LOC
- Linux kernel 2.2: ~2.5M LOC
- Hubble telescope: ~3M LOC
- Android core: ~12M LOC
- Army Future Combat Sys.: ~63M LOC
- Connected car: ~150M LOC
- Autonomous vehicle: ~300M LOC

Increasing Vulnerabilities

- Arduino, x86, ARM, CISC, RISC, server/desktop/laptop, desktop, laptop, smart phone, tablet

Increasing Threats

- In 2014:
  - Over 7,900 new vulnerabilities disclosed & catalogued
  - ~4,300 in Open Source, ~3,600 in commercial software

- NIST

Reference: Black Duck Software Knowledgebase, NVD

2012 - ARAMCO  
2014 - SONY  
2015 - UKRAINE
ROBOTICS PLATFORMS NOT EXEMPT

Sensors
- Integrity attacks (i.e. spoofing), e.g.
  - GPS PNT attacks
  - Lidar spoofing
- Availability attacks (i.e. Denial of Service)

GPS spoofing (Nighswander, 2012)
Lidar spoofing (Petit, 2015)

Communication
- Confidentiality attack – e.g. Traffic eavesdropping
- Inadequate key management/poorly implemented encryption algorithms
- Integrity attacks – e.g. Buffer overflow/remote code execution, code injection
- Availability attacks – Denial of Service/Jamming
- Over the air (OTA) software updates

Onboard processing
- Integrity - Unauthenticated commands
- Broad attack surface – Little to no IP Port/Protocol restrictions
- Availability attack against legitimate commands
- Close access attacks
  - USB ports
  - Maintenance laptops
  - Cell phone
  - Physical Insider

Other potential threat vectors
- Supply chain threat – e.g. FPGA bitstream files
- Software repositories
- Legacy components => frequency of patching & refresh of hw/sw
- Unique AI algorithmic vulnerabilities associated with autonomous systems

Controller
State Model
Planner
Open RF Comms
COMMS BUS (e.g. ETHERNET)
Sensor-1 (e.g. LIDAR)
Sensor-2 (e.g. GPS)
... Sensor-m
Actuator-1 (e.g. servo)
Actuator-2 (e.g. gripper)
... Actuator-n
## INSIGHTS

### General Principles
- Cybersecurity != Cyberspace defense—cannot defend everything – focus on ”key terrain”
- Must be able to detect, characterize, respond, and adapt within mission context

### Adversary actors
- Multiple ”online” personas associated with one physical identity
- Tactical actions derived from goals/intents
- Both parallel (e.g. reconnaissance, DDOS) and sequential (e.g. delivery/exploitation) action
- Cognitive, Logical, and Physical indicators

<table>
<thead>
<tr>
<th>Cyberspace Layer</th>
<th>Indicators</th>
<th>Detection Difficulty (Relative)</th>
<th>Adversary Cost to Change (Relative)</th>
</tr>
</thead>
</table>
| Persona/Cognitive | • Personas and Identities  
• Intent/Goals  
• Tactics, Tech., Procedures + C2  
• Social Presence and communication | Hard | Medium (more difficult after foothold is gained) |
| Logical | • Malware variants  
• IP addresses/TCP Ports  
• Configurations/Logs  
• File hashes | Low->Medium (depending on adversary sophistication) | Low |
| Physical | • Infrastructure  
• Computing nodes  
• Electromagnetic Spectrum  
• Geo-Location  
• Persona biometrics (key stroke, mouse patterns, facial recognition) | Medium | High (lower after foothold is gained) |
INSIGHTS

• General Principles
  ➢ Cybersecurity != Cyberspace defense--cannot defend everything – focus on ”key terrain”
  ➢ Must be able to detect, characterize, respond, and adapt

• Adversary actors
  ➢ Multiple ”online” personas associated with one physical identity
  ➢ Multiple tactical actions (derived from goals/intents) to achieve objectives
  ➢ Both parallel (e.g. reconnaissance, DDOS) and sequential (e.g. delivery/exploitation) action
  ➢ Cognitive, Logical, and Physical indicators

• Shortfall of expertise
  ➢ Well documented shortage of cyber expertise
  ➢ Combat units do not have cognitive resources to fight kinetic and non-kinetic fight simultaneously
  ➢ Demands some autonomy  *(but there is a complexity tradeoff)*

• Autonomous systems present new attack vectors
  ➢ Key benefit to autonomy – system’s ability to ”decide what to do next”
  ➢ Decision knowledge emerges from perception and memory – both subject to compromise

• Trustworthiness & Trust - Key obstacle to employment of autonomous systems

**Generation Gap Could Lead to a Cybersecurity Worker Shortage**

*Schools are scrambling to provide courses that emphasize cybersecurity, an element traditional computer science tracks have not included.*
**CONCEPTUAL APPROACH TRUSTWORTHY FRAMEWORK FOR AUTONOMY**

**Hypothesis:** Trustworthy framework for autonomy composed of three characteristics

<table>
<thead>
<tr>
<th>Trust Models*</th>
<th>Deterrence</th>
<th>Knowledge</th>
<th>Identification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ratnasignham, 1998</td>
<td>Cognitive</td>
<td>Emotional</td>
<td>Behavioral</td>
</tr>
<tr>
<td>Lewis &amp; Weigert, 1985</td>
<td>Habits</td>
<td>Passion</td>
<td>Policy</td>
</tr>
</tbody>
</table>

*From Wallace, 2007

**Trustworthiness & Trusted**

**Common traits:**
- Predictability
- Understanding
- Similarity
- Consequences
CONCEPTUAL APPROACH - TRUSTWORTHY FRAMEWORK FOR AUTONOMY

Hypothesis: Trustworthy framework for autonomy composed of three characteristics

- **Bounded**
  - Detect & characterize violations
  - Continuous validation & verification
  - AI algorithmic vulnerabilities & mitigations must be addressed (e.g. adversarial machine learning)

- **Explainable**
  - Support transparency through appropriate representations & processes
  - Model user state/comprehension
  - Multi-modal, adaptive, & interactive user interfaces

- **Trustworthiness & Trusted**
  - Consequences
  - Respond and adapt to violation of bounds
  - Fail-safe – reduced functionality
  - Resilient – Continue mission with reduced functionality (mission success may be less efficient and effective)

- **Fail-safe & Resilient**
  - Understanding
  - Similarity
**Challenges & Potential Approaches**

- **Bounded behavior – detect & characterize**
  - Behavioral meta-models (Wallace, 2007)
  - Monitoring and Validating Synthetic Behavior (Jones, 2015)
  - Top-down, Abductive Reasoning for Behavior Detection (Crossman, 2011)
  - Ethics (Arkin, 2012)
  - Safety Envelope for Security (Tiwari, 2014)

- **Cyber (?) – Research Gap**
  - Friendly Behavior Envelope

**Adversary Behavior Envelope**

- **Explainable - Support Transparency**
  - Episodic Memory (Nuxoll, 2007)
  - Model of User state/comprehension + multi-modal interfaces (Taylor, 2012)

- **Fail-Safe & Resilient - Respond and adapt – Research gaps**
  - What/Who makes decision to move to a fail-safe state?
  - What are the space of actions?

---

“Trust but verify”
- Army leadership philosophy
**Cyber Defense Battle Buddy Concept**

**Use Case (Friendly)**

1. $R_{1C}$, $R_{2C}$, $R_{3C}$ observe multiple $R_3$ connections to a.b.c.d/443 via logged connections
2. $R_{3C}$ directs collection of physical signal emissions emanating from $R_3$ to confirm/deny
3. $R_{1C}$, $R_{2C}$, $R_{3C}$ (majority) agree that $R_3$ has a boundary violation (transmitting to unknown IP) and recommend/decide on one of following actions (situation dependent: *cyberspace maneuver*)
   1. Block IP connections to a.b.c.d (via $R_3$ iptables)
   2. Repurpose $R_3$ as $R_{3C}$ (and vice versa) to enable communication to continue and observe
   3. Hunt for communicating process on $R_3$ and shut down
   4. Etc.

**Use Case (Adversary)**

1. Gain access to $R_3$ via remote code exploit (RCE) through RF inject into vuln. P2P software (e.g. a ROS Node)
2. Decrypt install binary and write to disk
3. Execute install to extract in-memory implant/backdoor
4. Send heartbeat to C2 server and receive instructions for rendezvous collection point; Remove install binary
5. (Persona through C2 server) recon file system for relevant plans
6. On order execute exfil to RP (repeat) – mission plans
7. On order wipe drive (destroy)
**Cyber Defense Battle Buddy Technical Approach**

**NOTES:**

1. General Purpose Processor (GPP) or embedded system with ability to partition address space
2. Hardware based hypervisor for efficiency and to support out-of-band processing.
3. VM subscripts (or more) – focused on the tactical behaviors to support synchronized kinetic + non-kinetic maneuver
4. VM subscripts – focused on behavior monitoring (communicate with other monitors preferable using out-of-band, non-operational link).
5. Tactical Behavior implementation for kinetic/non-kinetic maneuver and cyber monitor
**WHAT DOES SOAR HAVE TO DO WITH THIS APPROACH?**

**Procedural**
- Hierarchical control & reasoning
- Abductive reasoning (hypothesis testing)
- Transitions to fail-safe states (policies)

**Semantic**
- Adversary attack graphs (doctrinal templates)
- Compute network nodes and connections
- Friendly tools, techniques

**Working memory**
- Situational context - what is broader mission context?

**Symbolic Long-Term Memories**
- Procedural
- Semantic
- Episodic

**Symbolic Working Memory**
- Reinforcement Learning
- Chunking
- Semantic Learning
- Episodic Learning

**Spatial Visual System**
- Perception
- Action

**Episodic**
- Explaining behavior
- Reduce hypothesis search space (these are the indicators I looked for last time in this situation)

**SVS**
- Physical indicators (e.g. geo-location of threat vectors)
- Integration of kinetic/non-kinetic maneuver (in order to exploit through RF, must have transmitter within radius x)
EVALUATION — NONE

{SOME RESEARCH & EVALUATION QUESTIONS}

• What are the design space tradeoffs?
  - Number and types of monitoring agents?
  - Self-monitoring or group monitoring with voting (majority) algorithm
  - Soar controlling both tactical kinetic/non-kinetic behavior and cyber defense monitoring agents? If separate, how/when do they interact?
  - What is CPU overhead? Communications overhead?

• What cyber-related knowledge is most useful for detection?
  - Cognitive – are behavior envelopes sufficient for tracking adversary behavior?
  - Logic - OS/App logs, file hashes, security tools’ output
  - Physical emissions, spatial (e.g. geolocation) and temporal

• What are the unique vulnerabilities associated with AI systems? What are potential mitigation countermeasures?

• What is necessary for supporting infrastructure?
  - Modeling and simulation environment and tools to support development and experimentation
  - Physical platforms, space, and cyber/EW tools to support live experimentation
## Nuggets & Coal

<table>
<thead>
<tr>
<th>Nuggets</th>
<th>Coal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exploring Soar applicability in a new domain (Cyberspace)</td>
<td>No design, implementation, evaluation 😞</td>
</tr>
<tr>
<td>Exciting, explosive area</td>
<td>Unclear of right approach – much hype around AI and “cognitive” approaches</td>
</tr>
<tr>
<td>A lot of interest (+Work)</td>
<td>A lot of work</td>
</tr>
</tbody>
</table>