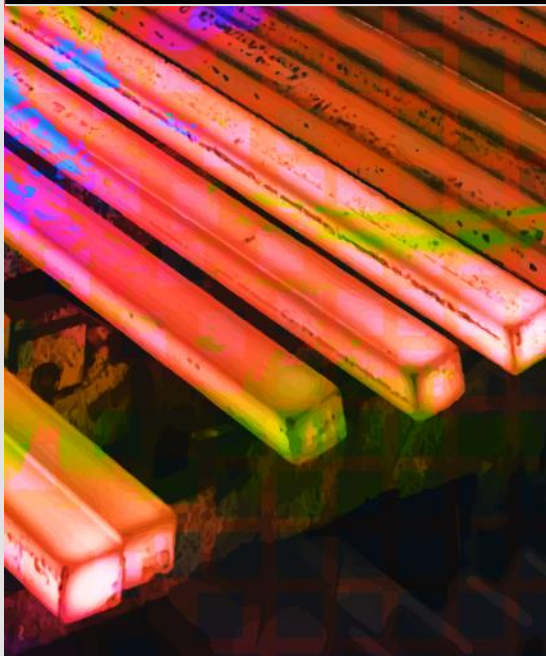
Carnegie Mellon University
Software Engineering Institute

Introducing the AI Security and Incident Response Team (AISIRT)



AUGUST 2024

Gregory Touhill, Brigadier General, USAF (ret)
Director, CERT Division

Document Markings

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM24-0174

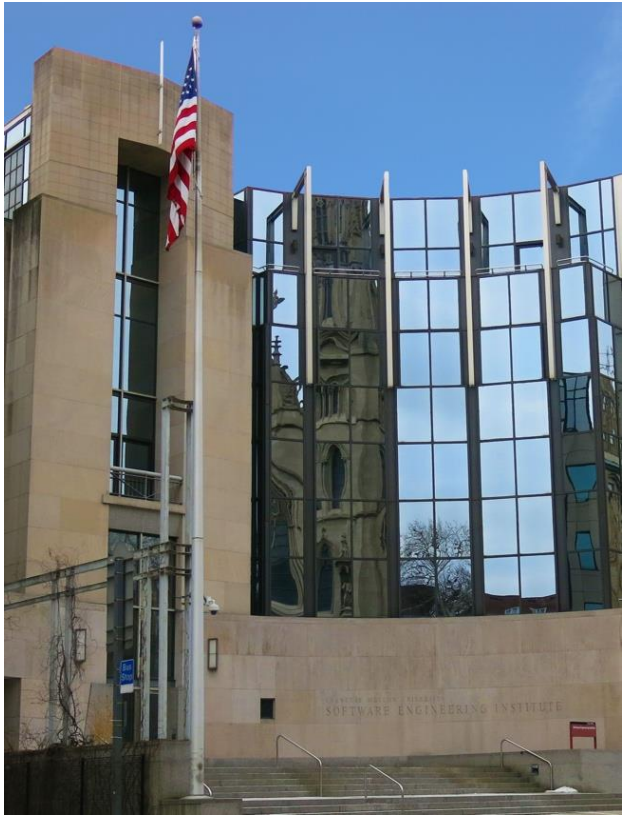
Carnegie Mellon University (CMU)

Pioneering discoveries on a global scale

- Leading-edge research global university turning disruptive ideas into mission successes
- 2023-2024 *U.S. News and World Report* rankings:
  - #1 in artificial intelligence, computer engineering, cybersecurity, management information systems, mobile/web applications, programming languages, software engineering, and quantitative analysis
  - #1 in overall computer science
- Creating inspired and inventive solutions through sponsored research, faculty and student engagement, executive education, licensing and tech transfer, start ups, and colocation

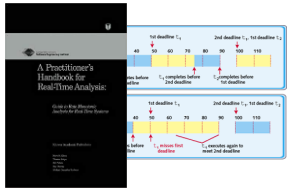
CMU Software Engineering Institute (SEI)

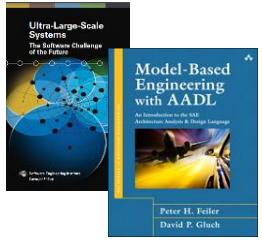
Bringing innovation to the U.S. Government



- Federally Funded Research and Development Center (FFRDC) chartered in 1984 and sponsored by the DoD
- Leader in researching complex software engineering, cybersecurity, and artificial intelligence (AI) engineering solutions
- Critical to the U.S. government's ability to acquire, develop, operate, and sustain software-enabled systems that are innovative, affordable, trustworthy, and enduring

CMU SEI: 40 years of Software Engineering Leadership





Repeatable delivery of platforms
Mid 1980s

Verification and validation of real-time scheduling
Early 1990s

Strategic design; reuse and evolution; measurement and analysis
Late 1990s

Ultra large scale systems research; Architectural Analysis and Design Language (AADL)
Mid 2000s

Adopting iterative development and acquisition practices
2010s

AI Engineering; AI Enabled Software Engineering & Cybersecurity
Today

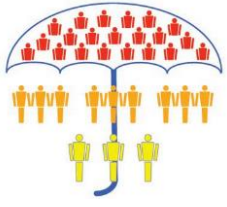
Late 1980s
Establishing a basis for software reuse

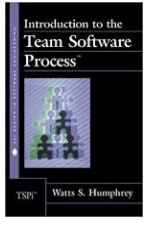
Mid 1990s
Evidence-based developer output

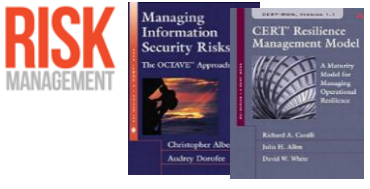
Early 2000s
Managing the operational risk of fielded systems

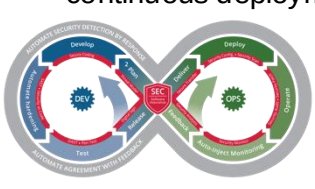
Late 2000s
SOA; secure design patterns; framework for architectural decisions

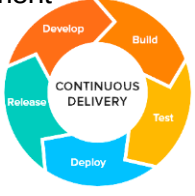
2020+
Lifecycle automation, Agile/DevOps/ DevSecOps; assuring complex systems; continuous integration/ continuous deployment











Artificial Intelligence is an Overnight Success

(69+ years in the making!)

Carnegie Mellon University Software Engineering Institute

A PROPOSAL FOR THE DARTMOUTH SUMMER RESEARCH PROJECT ON ARTIFICIAL INTELLIGENCE

J. McCarthy, Dartmouth College
M. L. Minsky, Harvard University
N. Rochester, I.B.M. Corporation
C.E. Shannon, Bell Telephone Laboratories

August 31, 1955

We propose that a 2 month, 10 man study of artificial intelligence be carried out during the summer of 1956 at Dartmouth College in Hanover, New Hampshire. The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. An attempt will be made to find how to make machines use language, form abstractions and concepts, solve kinds of problems now reserved for humans, and improve themselves. We think that a significant advance can be made in one or more of these problems if a carefully selected group of scientists work on it together for a summer.

The following are some aspects of the artificial intelligence problem:

1 **Automatic Computers**

If a machine can do a job, then an automatic calculator can be programmed to simulate the machine. The speeds and memory capacities of present computers may be insufficient to simulate many of the higher functions of the human brain, but the major obstacle is not lack of machine capacity, but our inability to write programs taking full advantage of what we have.

2. **How Can a Computer be Programmed to Use a Language**

It may be speculated that a large part of human thought consists of manipulating words according to rules of reasoning and rules of conjecture. From this point of view, forming a generalization consists of admitting a new word and some rules whereby sentences containing it imply and are implied by others. This idea has never been very precisely formulated nor have examples been worked out.

3. **Neuron Nets**

How can a set of (hypothetical) neurons be arranged so as to form concepts. Considerable theoretical and experimental work has been done on this problem by Uttley, Rashevsky and his group, Farley and Clark, Pitts and McCulloch, Minsky, Rochester and Holland, and others. Partial results have been obtained but the problem needs more theoretical work.

4. **Theory of the Size of a Calculation**

If we are given a well-defined problem (one for which it is possible to test mechanically whether or not a proposed answer is a valid answer) one way of solving it is to try all possible answers in order. This method is inefficient, and to exclude it one must have some criterion for efficiency of calculation. Some consideration will show that to get a measure of the efficiency of a calculation it is necessary to have on hand a method of measuring the complexity of calculating devices which in turn can be done if one has a theory of the complexity of functions. Some partial results on this problem have been obtained by Shannon, and also by McCarthy.

5. **Self-Improvement**

Probably a truly intelligent machine will carry out activities which may best be described as self-improvement. Some schemes for doing this have been proposed and are worth further study. It seems likely that this question can be studied abstractly as well.

6. **Abstractions**

A number of types of ``abstraction'' can be distinctly defined and several others less distinctly. A direct attempt to classify these and to describe machine methods of forming abstractions from sensory and other data would seem worthwhile.

AI is in Our Blood at Carnegie Mellon

Carnegie Mellon University Software Engineering Institute

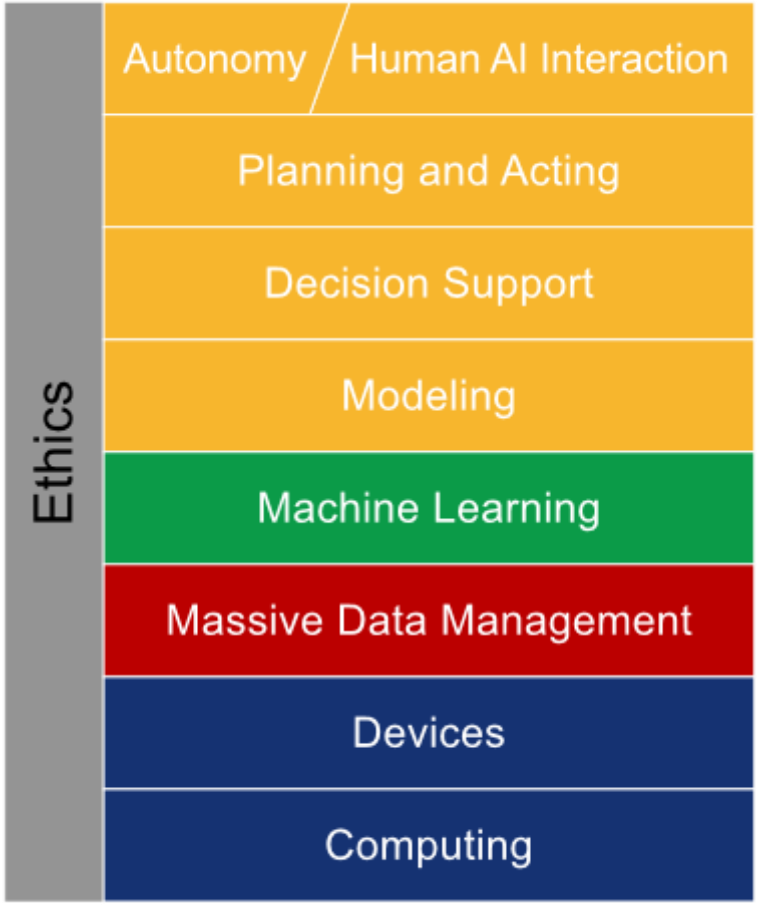
- Home of the DoD-sponsored Software Engineering Institute
- CERT: the birthplace of cybersecurity
- US News & World Report #1 ranking in Computer Science, Cybersecurity, AI, and Programming
- Designated National Center of Academic Excellence in Cyber Research
- Home of the National Robotics Engineering Center
- DoD's Center for Calibrated Trust Measurement and Evaluation (CaTE)
- Headquarters of the Army Artificial Intelligence Integration Center
- Self-Driving Cars
- Speech Recognition
- Facial recognition software
- Advanced sensor design and application
- Internet of Things
- Machine learning
- Data science and analytics
- AI ethics, Responsible AI, Adversarial Machine Learning, Trustworthy Computing, and more…

Why Do We Need An AISIRT?

Carnegie Mellon University Software Engineering Institute

- Military, government, critical infrastructure and business operations are increasingly reliant on AI Systems
- AI Systems and their components are under attack, susceptible to confidentiality breaches, subject to defects that can adversely affect mission operations, and other kinds of failures
- AI Systems are not always created and operated under “Secure by Design” principles, DevSecOps techniques and other best practices
- Tactics, techniques, procedures, and tooling to test, analyze, monitor, audit, and holistically protect AI Systems are either non-existent or in their infancy
- Vulnerability Management coordination for vulnerabilities, defects, incident reporting, and other cyber-related issues are poorly defined

Attacks on AI are coming from multiple vectors



Adversarial AI Attacks:
Universal and Transferable Adversarial Attacks on Aligned Language Models:
https://arxiv.org/abs/2307.15043

Adversarial ML Attacks:
Keras 2 Lambda Layers Allow Arbitrary Code Injection in TensorFlow Models:
https://kb.cert.org/vuls/id/253266

Traditional Cyber Attacks:
Graphics Processing Unit (GPU) kernel implementations susceptible to memory leak:
https://kb.cert.org/vuls/id/446598

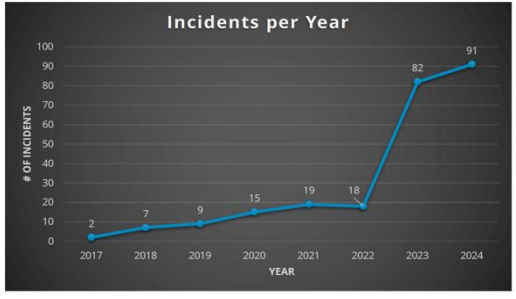
AI enables deepfakes

(so guess who created a deepfake detection capability?)

Issue definitions

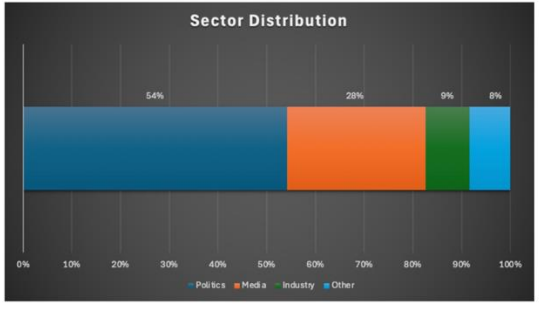
Issue	Definition
Mis/disinformation	The use of deepfakes to intentionally or unintentionally spread false information.
Fraud	The use of deepfakes to impersonate high-ranking individuals, employees, or customers for deception leading to financial gain.
Defamation	The use of deepfakes to target a specific person leading to loss of reputation.
Exploitation	The use of deepfakes that intentionally or unintentionally causes harm to a specific group of people.
Identity theft	The use of deepfakes to impersonate someone for personal gain.

Results: Year



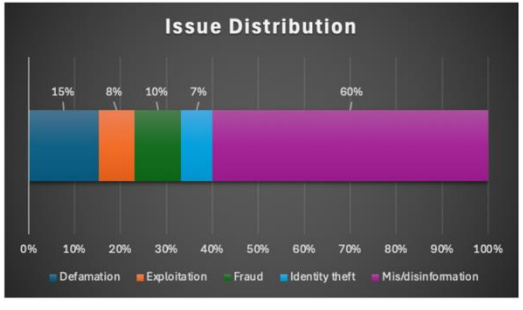
Projection indicates continued upward trend

Results: Sector



Most incidents occurred in the political sector

Results: Issue



Mis/disinformation was the most common type of issue, followed by defamation

AISIRT Focus and Planning Assumptions

Carnegie Mellon University Software Engineering Institute

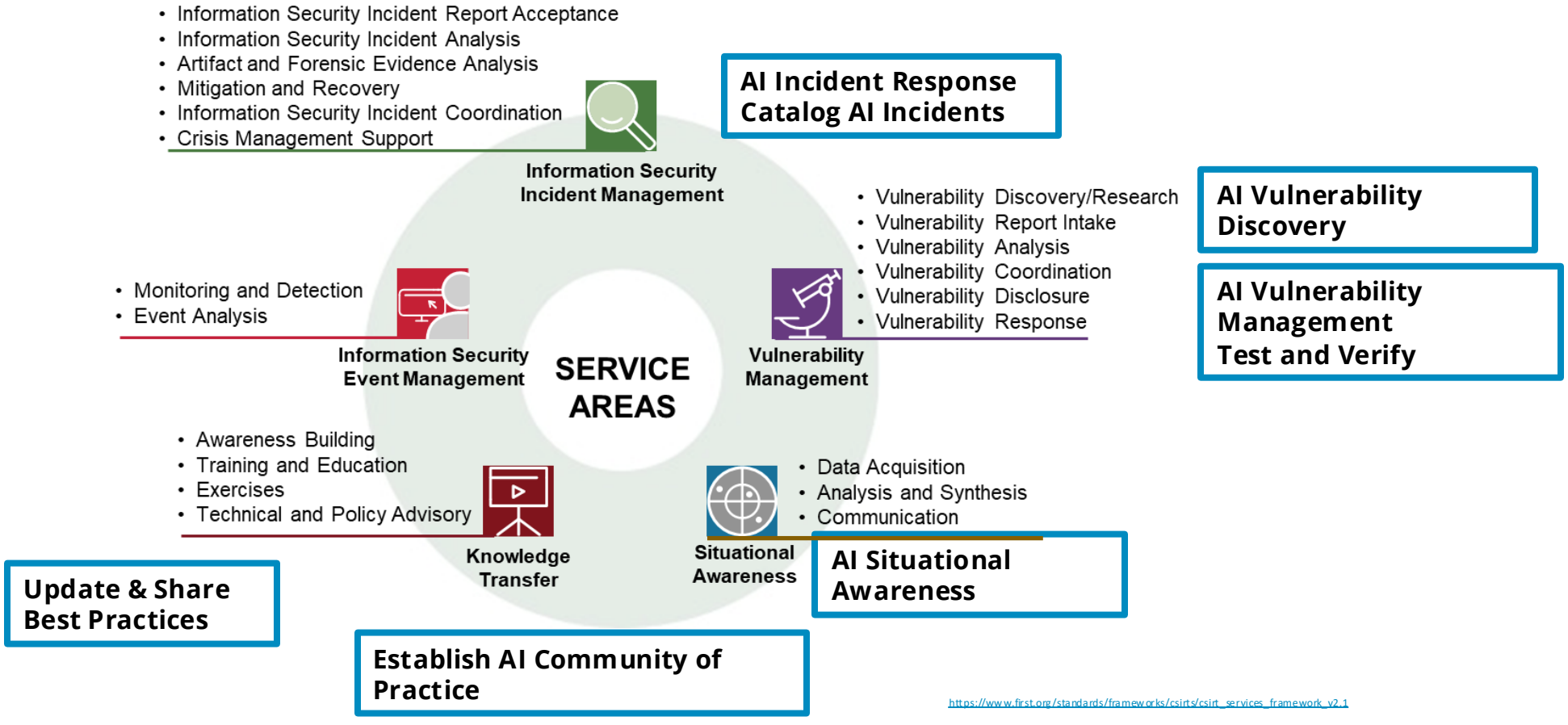
Focus

- Identification, understanding, and mitigation of vulnerabilities for AI systems of interest to and in use by defense and national security organizations
- Vulnerabilities include traditional cybersecurity, adversarial machine learning, and joint cyber-AI attacks

Planning Assumptions

- Initial focus on defense and national security related incidents can yield benefits to national critical infrastructure
- Joint effort across Carnegie Mellon University and CERT Division's partner network
- Leverage existing rules of engagement from cyber incident response where applicable (e.g. information sharing, after action reporting)

An AISIRT in the context of a CSIRT



https://www.first.org/standards/frameworks/csirts/csirt_services_framework_v2.1

AISIRT Mission Essential Functions and Tasks

Carnegie Mellon University Software Engineering Institute

AI Incident Response

- Root Cause Analysis
- Mitigation
- Catalog AI Incidents

AI Vulnerability Discovery

- Passive: Reports from government, industry, and academia
- Active: Monitoring and red-teaming of defense and national security related systems
- Targeted: Based on mission-centered threat intelligence

AI Situational Awareness

- Monitoring: Develop capacity for real-time list of alerts, events and risks
- Communication: Synthesis of risks and communicating risks to U.S. Department of Defense

AISIRT Mission Essential Functions and Tasks

Carnegie Mellon University Software Engineering Institute

AI Vulnerability Management

- Prioritization: Leverage Software Engineering Institute-created Stakeholder-Specific Vulnerability Categorization (SSVC) process
- Test and Verify: Reproduction of exploits to quantify severity and impact (if able)
- Identify Constraints based on infrastructure, data access, etc.
- Coordination: Reach out to vendors to address
- Disclosure: Raise community awareness, provide mitigations & recommendations to reduce severity and impact

Update and Share Best Practices, Standards, and Guidelines for defense and national security organizations

- Assist industry in achieving secure by design and secure by default vision
- Improve the state of the art / state of the practice

Establish an AI Community of Practice across academia, industry, defense and national security organizations, and legislative bodies

AISIRT Composition

Create initial incident response team using CERT Coordination Center (CERT/CC) model

- Team lead who can translate the technical aspects in an understandable way to affected parties
- System/database administrator
- Network engineer
- AI/ML practitioner
- Threat intelligence researcher
- On-call specialists from Carnegie Mellon University and other trusted industry/academic partners as needed

Emerging Lessons Learned from AISIRT

Carnegie Mellon University Software Engineering Institute

- AI vulnerabilities are cyber vulnerabilities
- AI vulnerabilities are occurring throughout the entire system
- Cybersecurity processes should continue to evolve to support AI
- AI systems are different than today's traditional IT in several interesting ways
  - Good luck replicating issues in Generative AI
  - Complexity in AI systems complicates triage, diagnostics, and troubleshooting
- Tools to identify vulnerabilities aren't there yet
- Need for secure development training (i.e. DevSecOps) tailored for AI developers
- Red team pentesting of AI systems throughout the development cycle can identify material weaknesses early in the development cycle
- Lack of AI technical understanding leads to increased organizational risks

Leveraging the AISIRT

Carnegie Mellon University Software Engineering Institute

Contact the AISIRT

- info@sei.cmu.edu
- +1.412.268.5800

Report Vulnerabilities

- https://www.kb.cert.org/vuls/vulcoordrequest/

Consult our Publications

- https://www.sei.cmu.edu

Share Your Experiences

- Call us, write us, invite us for a visit

Upgrade Your AI and Cyber knowledge through our CMU Courses:

- Work with us to create one tailored to your needs
- Leverage existing courses: https://www.heinz.cmu.edu/programs/executive-education/

Questions?