Center for Trustworthy Scientific Cyberinfrastructure

The mission of CTSC is to provide the NSF community with a coherent understanding of cybersecurity, its importance to computational science, and the resources to achieve and maintain an appropriate cybersecurity program.

CENTER FOR APPLIED CYBERSECURITY RESEARCH
INDIANA UNIVERSITY
Pervasive Technology Institute

NCSA

PITTSBURGH SUPERCOMPUTING CENTER

THE UNIVERSITY of WISCONSIN MADISON

NSF

trustedci.org

A bit of history

Oct 2012: CTSC began; 3-year NSF project
Jan 2016: NSF Cybersecurity Center of Excellence (CCoE)

trustedci.org

SGCI & CTSC co-fund a 50% FTE to help address cybersecurity for the gateways community. (Formally, part of SGCI's Incubator team)

CTSC: Engaging with NSF projects

http://trustedci.org/engagedcommunities/

~19 engagements so far (mid-2016)

Security needs/goals are different for each

- Range: [policy, technical]
- Final Report(s) (available at above URL)
- *describe a few engagements: next slides*

Formal engagement application process*:
http://trustedci.org/application/

* Not necessarily relevant for gateways

NSF Large Facilities: LIGO, Ice Cube, LSST,…

- Federated Identity
- Risk Assessments
- Review architecture & operations
- Security Policies (provide templates)

Wildbook

CTSC engaged with Wildbook and its Image-Based Ecological Information System (IBEIS) project to design and prototype single sign-on (SSO) and role-based access control (RBAC).

http://blog.trustedci.org/2016/10/ibeis.html

Software projects: Pegasus, SciGaP, Globus, perfSONAR

- Code Reviews (including FPVA*)
- Static Analysis
- Reviewing/testing security of new functionality, e.g., file sharing

* http://research.cs.wisc.edu/mist/includes/vuln.html

Gateways: Learning from the past…

- Duplication of effort: software, etc.
- Identity and Access Management challenges (e.g. community accounts, but individual accounting)
- Resources evolve (e.g., clouds, mobile)
- Protocols evolve (e.g., REST, OAuth)
- Security concerns evolve

Preparing for the future…

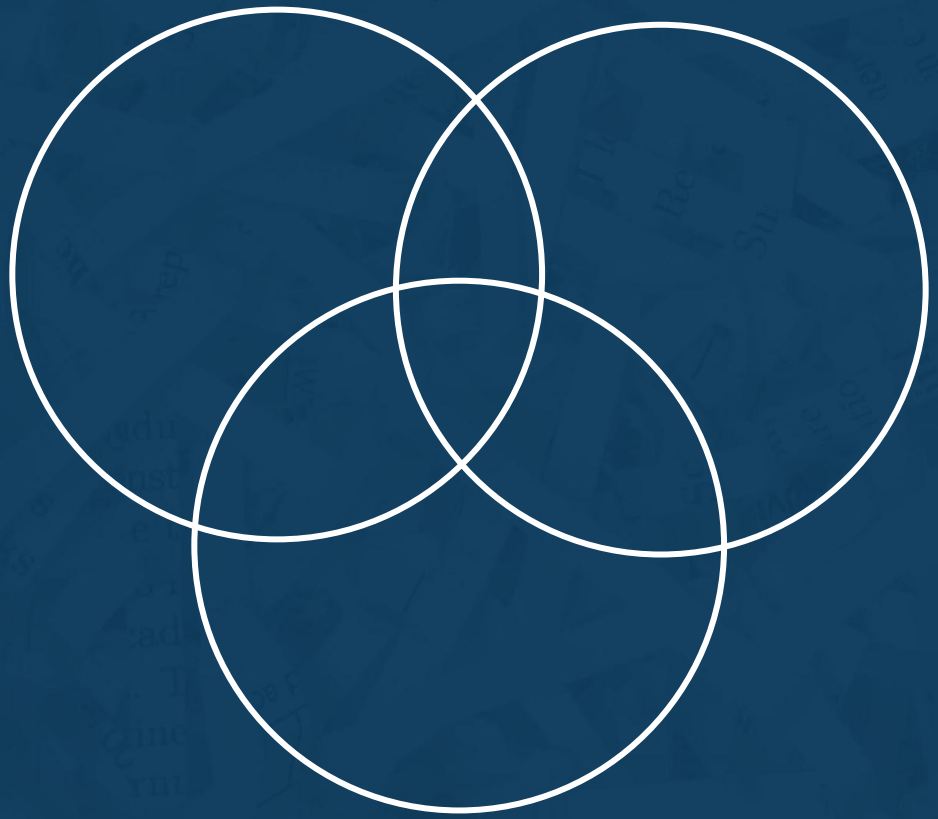
- Trying to avoid duplication of effort
  - SGCI, SciGaP, Globus, CILogon, CTSC, …
- Prepare/adapt to evolving resources, protocols, programming languages
- CTSC-related:
  - visit trustedci.org ; join mailing list(s)
  - check out resources: IAM, software security, risk profiles, security policy templates, etc.

NSF "CI Framework for 21st century" (CIF21)

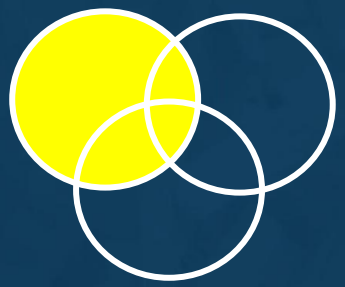
Software is fundamentally computer code. It can be delivered to end users in multiple formats, ranging from an archive that a user downloads and builds to an executable or a service running on a remote system to which a user connects. Especially at large scale, software is generally difficult to design, implement and then maintain, and the software needed by the science, engineering, and education communities is particularly complex. Software must be reliable, robust, and secure; able to produce trustable and reproducible scientific results; …

http://www.nsf.gov/pubs/2012/nsf12113/nsf12113.pdf

Software Security



CTSC has a thrust in each.



Software Assurance

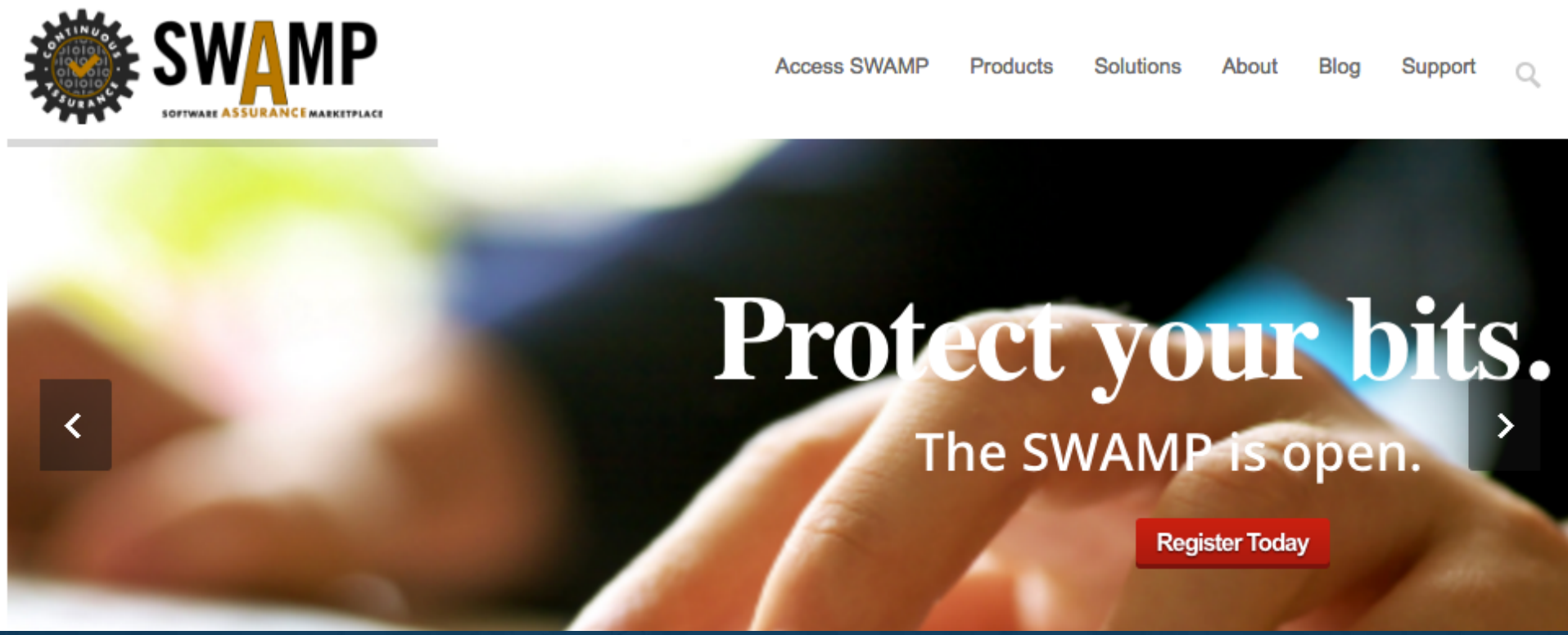
#1) SwA is the level of confidence that software is free from vulnerabilities, either intentionally designed into the software or accidentally inserted at any time during its life cycle, and that the software functions in the intended manner.

https://samate.nist.gov/Main_Page.html

#2) The processes (e.g., secure coding, static analysis) that help improve this level of confidence.

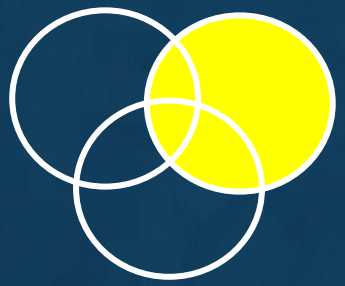
http://trustedci.org/trainingmaterials
http://sc16.supercomputing.org/presentation/?id=tut112&sess=sess223

Static analysis as a service: SWAMP



https://continuousassurance.org/

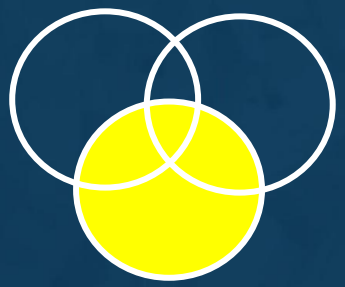
https://www.mir-swamp.org/#tools/public
https://continuousassurance.org/swamp-in-a-box/



Situational Awareness

Being aware of software vulnerabilities and how they might affect a user community. Offering advice on how to patch or update vulnerable software.

http://trustedci.org/situational-awareness
http://blog.trustedci.org/2016/08/situational-awareness.html



Secure Software Engineering

- Instill security awareness in software engineers - developers and testers.
- Educate them in appropriate processes, practices, and tools.
- Help deliver and maintain secure software over its entire lifecycle.

http://trustedci.org/trainingmaterials/

Secure Software Engineering Best Practices

- Repositories
- Testing
- Static Analysis
- Vulnerability Management
- Release & Delivery
- Coding/Project Tools
- Documentation

Summary

- CTSC is eager to engage with the science gateways community to develop more reliable, robust, and secure software.
- Please ask questions and join in discussions at:
  http://trustedci.org/ctsc-email-lists/
- Check out online resources at trustedci.org

Thank You

Questions?

trustedci.org
@TrustedCI

We thank the National Science Foundation (grants ACI-1547611 and ACI-1547272) for supporting our work.

The views and conclusions contained herein are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the NSF.