## Program Analysis and Verification on Trusted Platforms (PAVeTrust) Workshop

This workshop will be held online Monday, **December 6, 2021**, in conjunction with the Annual Computer Security Applications Conference (ACSAC).

**Statement on COVID-19.** This workshop and ACSAC is being held *virtually* this year. More information is available at the conference site, https://www.acsac.org/

Further details about the workshop can be found at the workshop website:
https://www.acsac.org/2021/workshops/pavetrust/

# Workshop Program

**All times are GMT**

**8:45-9 Opening remarks by Bill Roscoe (University of Oxford & The Blockhouse Technology Limited, UK)**

**9-10 Invited talk 1: "Formal Foundations for SCONE attestation and Intel SGX Data Center Attestation Primitives" by Christof Fetzer and Muhammad Usama Sardar (TU Dresden, Germany)**

Abstract:
"SCONE is a platform that enables the transformation of native services into confidential services.
A confidential service is automatically attested and verified by SCONE. Only if a service satisfies its security policy, it receives its secrets. Using mutualTLS, one can, for example, ensure that only verified services belong to the same application are permitted to communicate with each other.

SCONE supports both Intel Attestation Service as well as Intel DCAP. The Intel Data Center Attestation Primitives (DCAP) is a third-party attestation service to enable data centers to create their own attestation infrastructures. These services address the availability concerns and improve the performance compared to the remote attestation based on Enhanced Privacy ID (EPID).

The lack of formal proof for DCAP might cause security concerns. To fill this gap, we propose an automated, rigorous, and sound formal approach to specify and verify the remote attestation based on Intel SGX DCAP under the assumption that there are no side-channel attacks and no vulnerabilities inside the enclave. In the described approach, the data center configuration and operational policies are specified to generate the symbolic model, and security goals are specified as security properties to produce verification results. The evaluation of non-Quoting Verification Enclave-based DCAP indicates that the confidentiality of secrets and data integrity is preserved against a Dolev-Yao adversary in this technology."

**10:10-10:50 Paper 1: "Towards Leakage-Resistant Machine Learning in Trusted Execution Environments" by Mukesh Tiwari (University of Melbourne, Australia)**

**11:00-12:00 Invited panel: "Securing TEEs with Verification: Distant dream or a reality?" moderation by Shweta Shinde (ETH Zurich, Switzerland) and panelists are Anitha B Gollamudi (Yale University, US), Karim Eldefrawy (SRI International, US), and Aquinas Hobor (National University of Singapore, Singapore)**

**12:10-12:50 Paper 2: "To verify or tolerate, that’s the question" by Inês Gouveia, Marcus Völp, Muhammad Sakr, and Rafal Graczyk (University of Luxembourg, Luxembourg)**

**13:30-14:30 Invited talk 2: "The trusted verification of confidential code" by Bill Roscoe (University of Oxford & The Blockhouse Technology Limited, UK)**

Abstract:
"I will introduce the concept of the Transparency Centre, which enables a regulator or customer to verify or analyse executable code without having access to all representations of that code that needed for the analysis. So for example customers can, without trusting the vendor, know that the object code they have was compiled from sources that have been verified to meet a specification. I will review our experimental implementations and potential applications."

**14:40-15:20 Paper 3: "Confidential Computing and Related Technologies: A Review" by Muhammad Usama Sardar and Christof Fetzer (TU Dresden, Germany)**

**15:30-16:30 Invited talk 3: "Towards democratizing secure enclave programming" by Guido Salvaneschi (University of St.Gallen, Switzerland)**

Abstract:
"Secure enclaves, like Intel SGX, provide means to process data securely on third-party cloud infrastructure with little or no performance overhead.
Developing software that takes advantage of a secure enclave requires, however, to explicitly deal with a number of low-level details such as dedicated IO, custom syscalls, and hard memory constraint, resembling the expertise needed for system programming, rather than applications.
We discuss recent research that provides developers a more friendly approach to enclave programming, including our Java language extension JE.
Finally, we outline our vision of a programming framework that brings secure enclave programming at the fingertips of application developers."

**16:30-17:00 Wrap-up discussion and concluding remarks by Bill Roscoe (University of Oxford & The Blockhouse Technology Limited, UK)**

**Sponsored by**

The Blockhouse
Technology Ltd
www.tbtl.com