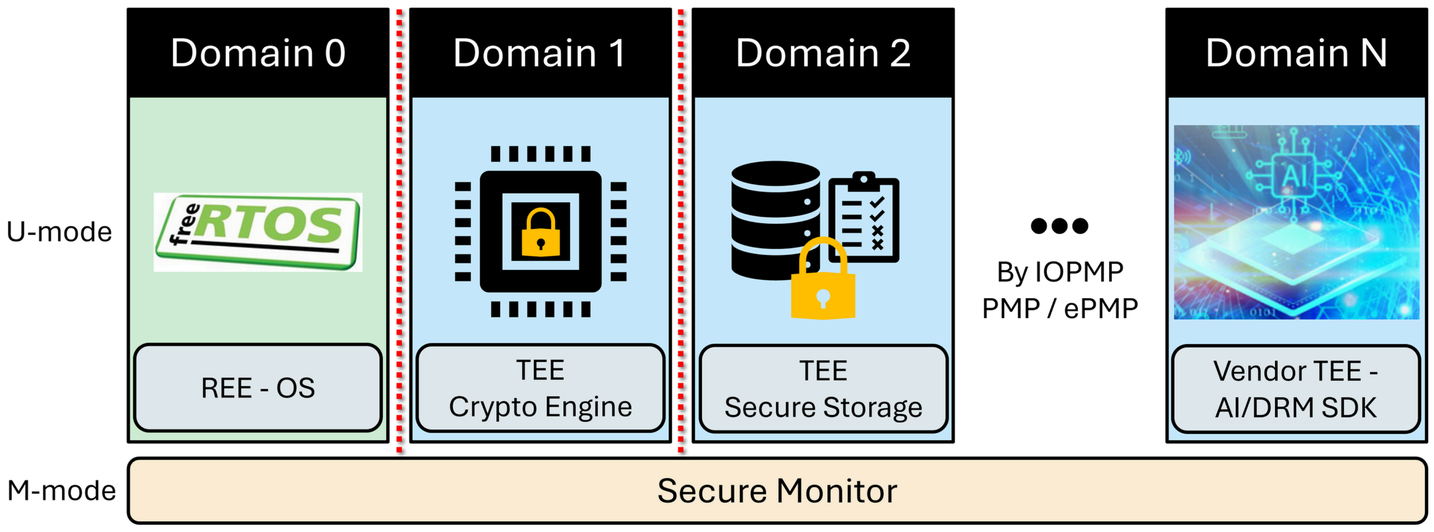
Secure Domain-Specific Debugging on an MCU

Alvin Che-Chia Chang and Paul Shan-Chyun Ku

Andes Technology

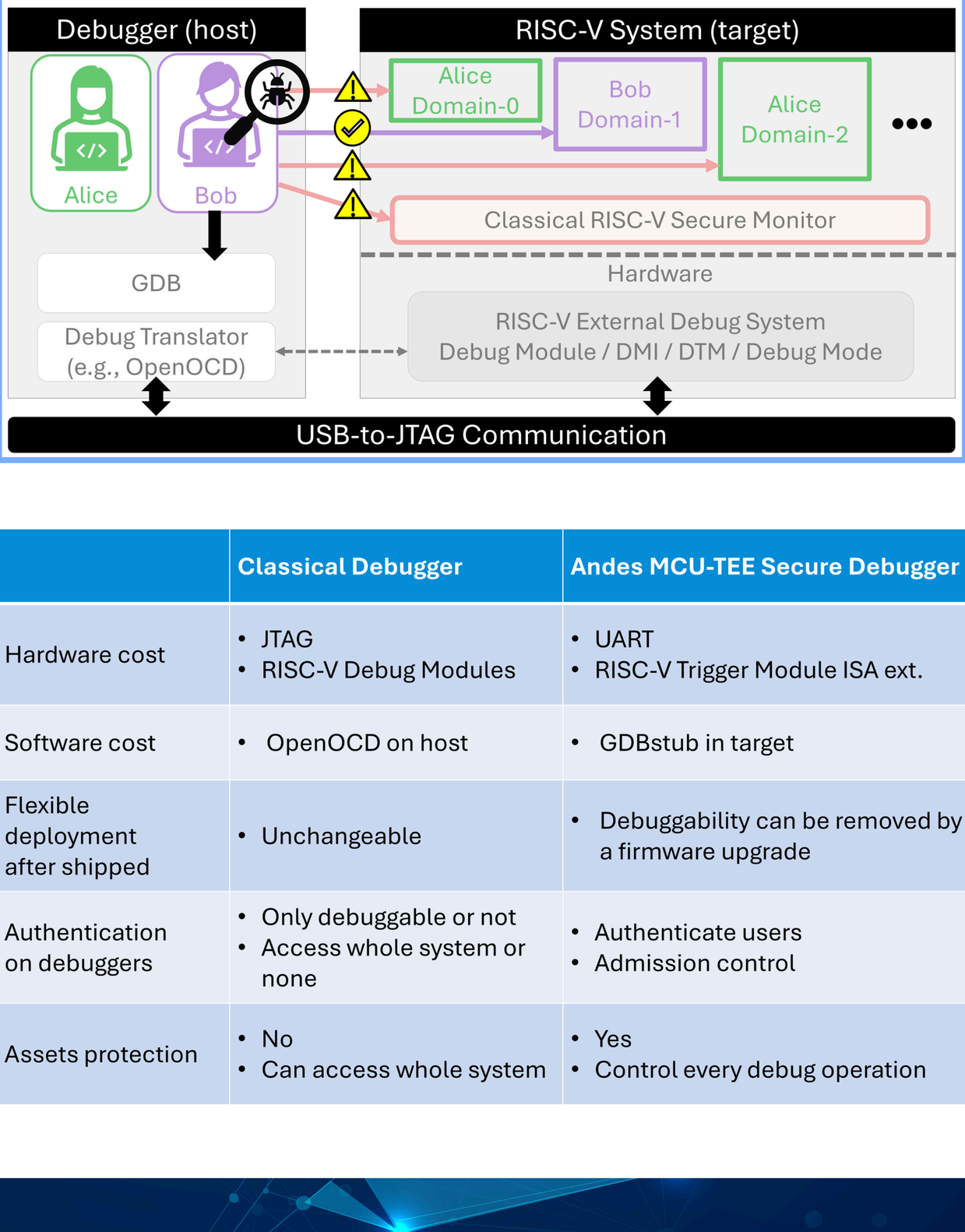
Execution Environment Isolation



- Create multiple Trusted Execution Environments (TEE)
- Isolate EE's resources (memory and interrupt)
- Protect multiple vendors' privacy

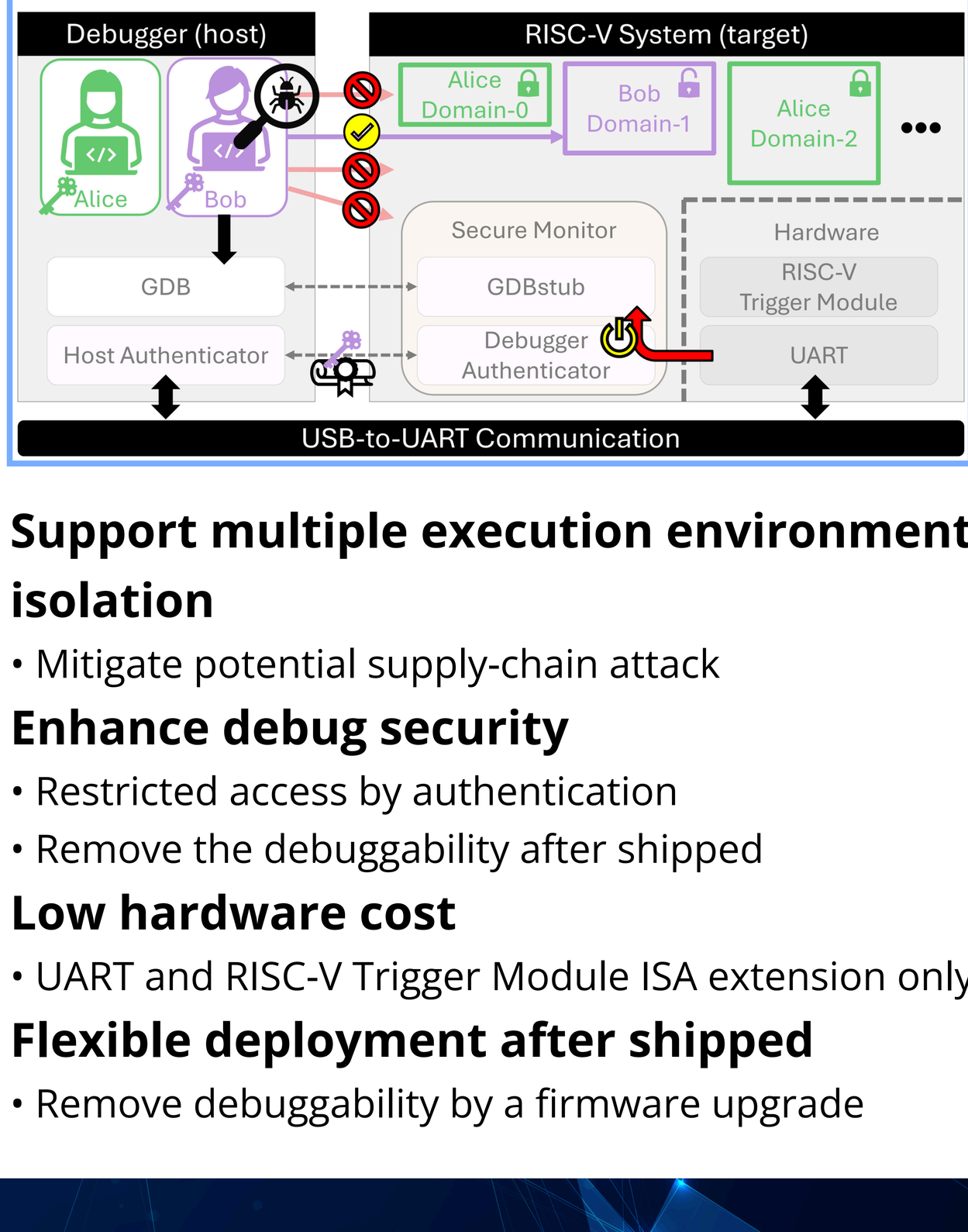
Problem of classical debugger: Vulnerable to supply chain attack

- Supply chain can snoop the whole system



Andes MCU-TEE's solution: Controls debugger's access

- Authenticate users and control admission
- Protect proprietary assets



	Classical Debugger	Andes MCU-TEE Secure Debugger
Hardware cost	• JTAG • RISC-V Debug Modules	• UART • RISC-V Trigger Module ISA ext.
Software cost	• OpenOCD on host	• GDBstub in target
Flexible deployment after shipped	• Unchangeable	• Debuggability can be removed by a firmware upgrade
Authentication on debuggers	• Only debuggable or not • Access whole system or none	• Authenticate users • Admission control
Assets protection	• No • Can access whole system	• Yes • Control every debug operation

Support multiple execution environment isolation

- Mitigate potential supply-chain attack

Enhance debug security

- Restricted access by authentication
- Remove the debuggability after shipped

Low hardware cost

- UART and RISC-V Trigger Module ISA extension only

Flexible deployment after shipped

- Remove debuggability by a firmware upgrade