

DARPA's 5G Portfolio

Tejas Patel, Program Manager
Information Innovation Office

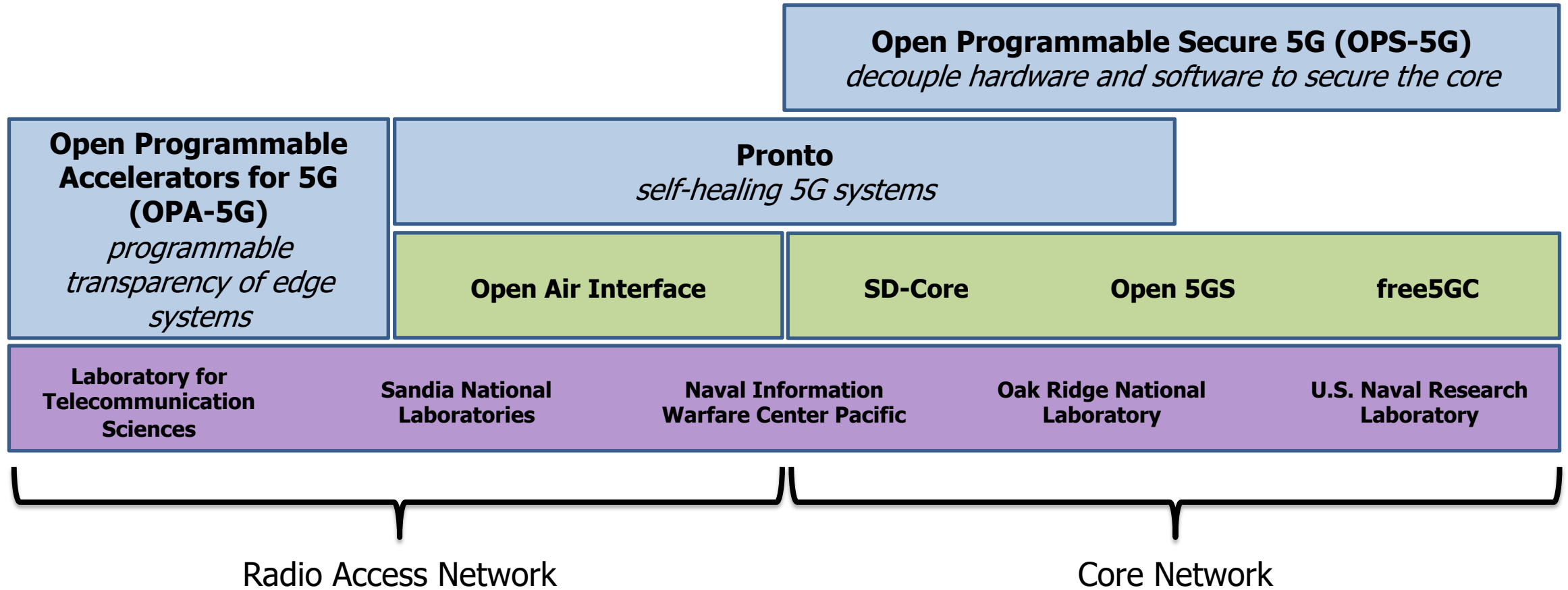
Briefing prepared for the Carnegie Mellon University Software Engineering Institute
Workshop on Software Systems at the Edge



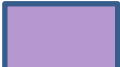
November 30, 2022



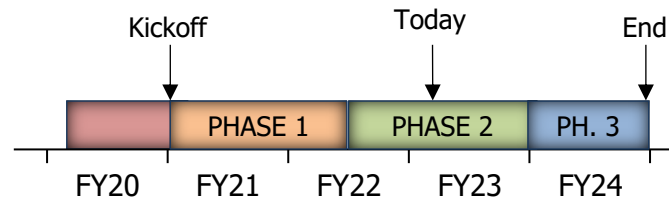


Goal: Develop secure 5G infrastructure



 DARPA efforts  Open source  Government partners

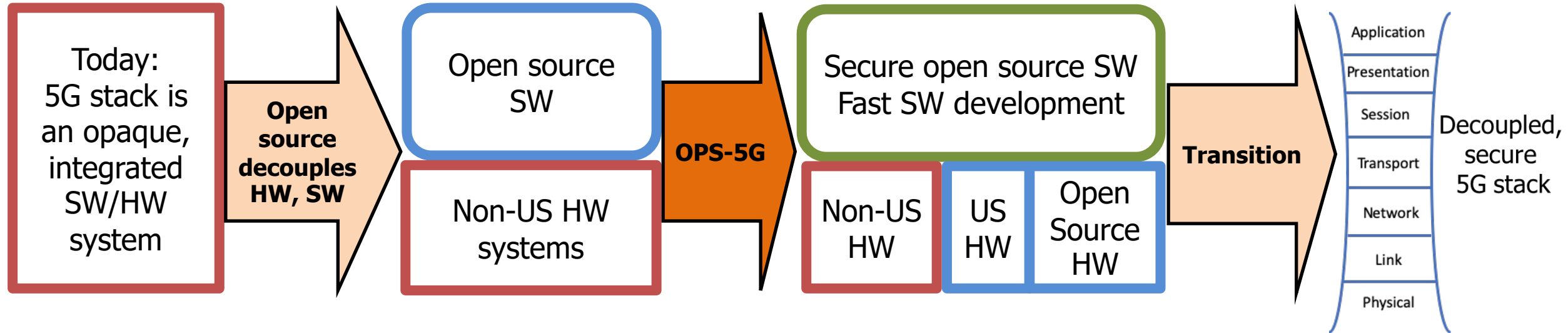
Open Programmable Secure 5G (OPS-5G)





OPS-5G vision

Create open source software and systems enabling secure 5G and subsequent mobile networks



Challenges

- 5G flexibility/programmability opens vulnerabilities
- Sharing of network resources with unknown parties make US network traffic susceptible to side channel attacks
- Tremendous growth of security-free 5G-enabled IoT devices create large-scale security issues

Approach

- Use programmable elements of 5G for defense
- Adapt cryptographic approaches to secure network resources shared by unknown entities
- Implement scalable SWaP-conscious encryption primitives and security protocols

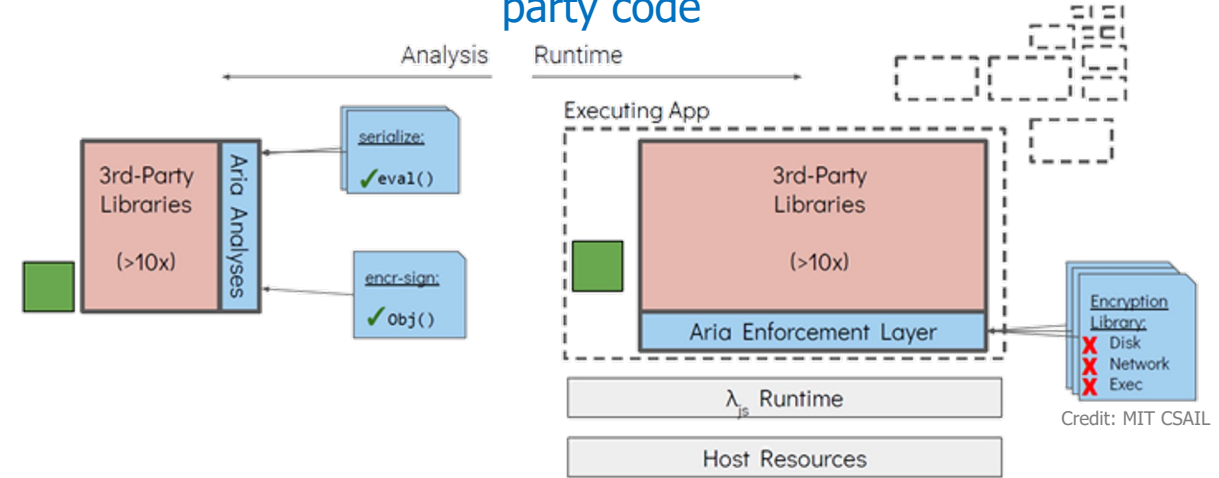


MIT: Capability-safe subset of JavaScript

Approach:

- Define a capability-safe subset of JavaScript (J_{CS})
- Verify code using security toolchain
 - If program in J_{CS} : statically check permissions using Mir
 - If program outside of J_{CS} : dynamically enforce permissions at runtime (dynamic instrumentation system Lya)

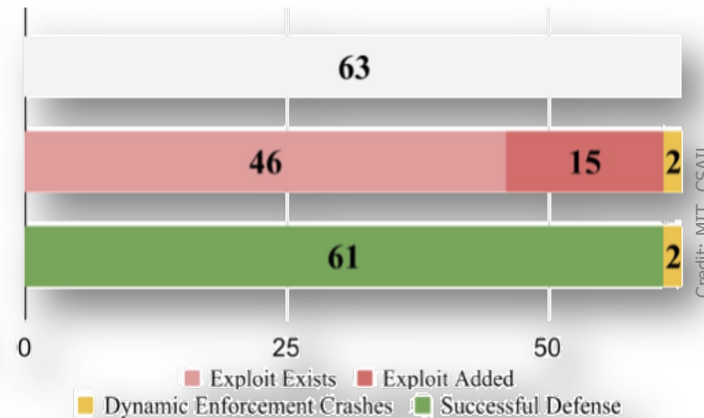
Inference and enforcement of policies around third-party code



Security evaluation across Snyk's JavaScript VulnDB



Malicious libraries



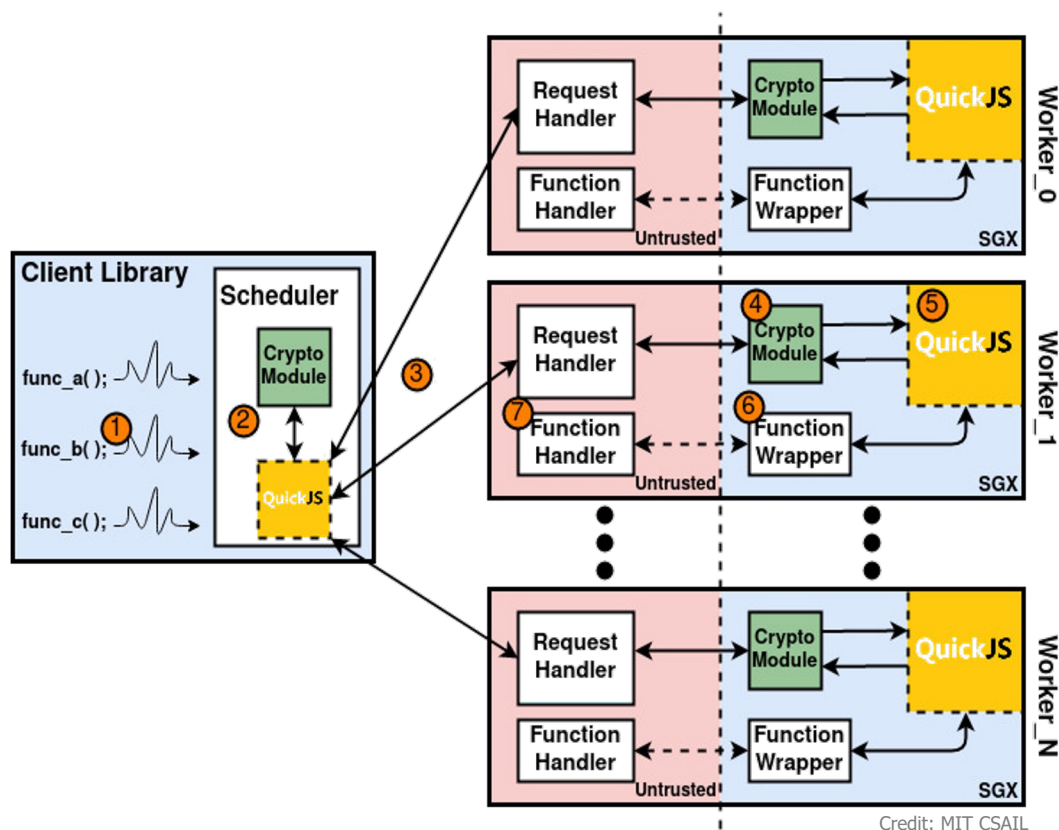
RWX benign-but-buggy libraries

Generality/expressiveness of approach

Class	Node-chat		Official-portfolio	
	Instances	LoC diff per instance	Instances2	LoC diff per instance2
Prototype-Chain Method Skipping	357	0	91	0
Runtime Metaprogramming	8	1	9	1
Special Built-in	10	0	6	0
Bug: Direct import invocation	41	2	41	2

Total LoC: 57,040
563 statements (<1%) not handled automatically

Credit: MIT CSAIL



Secure offloading from client to workers:

- Client identifies function for offloading
- Client encrypts/signs request (data and code)
- Client forwards request to a worker
- Worker forwards request to SGX
- Worker decrypts request and executes function
- Worker encrypts/signs results

Average duration (in seconds) and battery consumption (by battery percentage) by input size (in bytes) for the encrypt/sign operations

Input size	Local execution		Offloaded execution	
	Duration1	Battery1	Duration2	Battery2
16	0.0302	0.073294	0.0797	0.078956
32	0.0306	0.008434	0.0782	0.016451
64	0.0323	0.004799	0.0806	0.008392
128	0.0388	0.003697	0.0825	0.004695
256	0.0479	0.002757	0.0840	0.003070
512	0.0672	0.001770	0.0847	0.002447
1K	0.1093	0.002169	0.0862	0.002008
2K	0.1773	0.002014	0.0813	0.002406
4K	0.3332	0.003160	0.1134	0.003022
8K	0.5954	0.003310	0.1136	0.000120
16K	1.2693	0.006416	0.1506	0.003374
32K	2.3706	0.009455	0.2283	-0.001098
64K	4.9341	0.013232	0.3845	0.001883
128K	9.8098	0.025901	0.6468	-0.001459
256K	18.8672	0.059975	1.0996	0.000555
512K	39.9212	0.689173	2.0907	0.002643
1M	78.1807	1.616290	4.5572	0.120933

Credit: MIT CSAIL

Results summary:

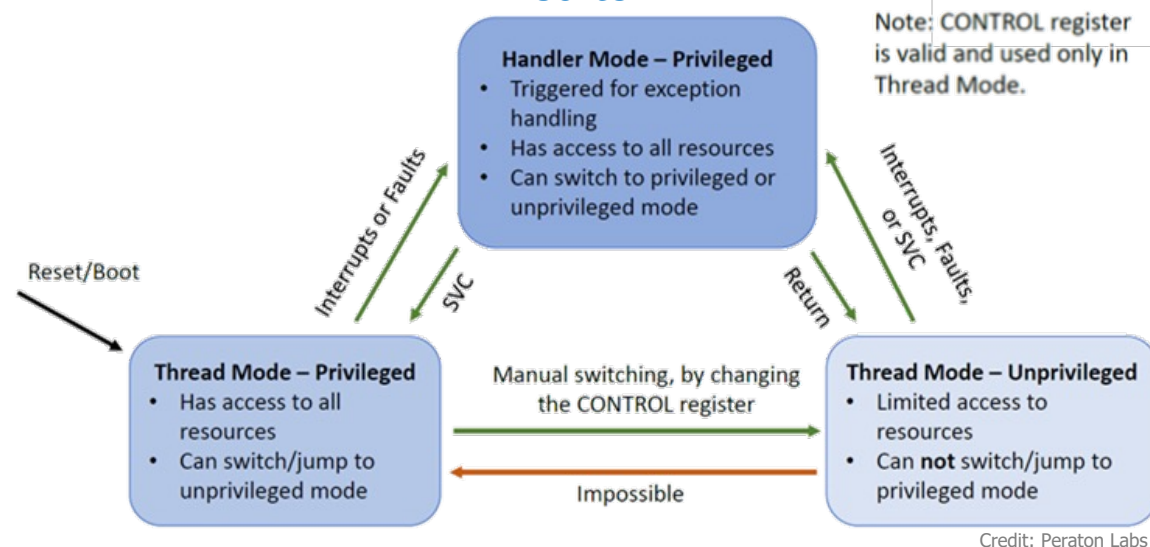
- Solution saves battery (up to ~13x battery savings) and execution time (up to ~20x time savings)
- Demonstrations of computational offload with good results at larger file sizes; performance dependent on operation and input size



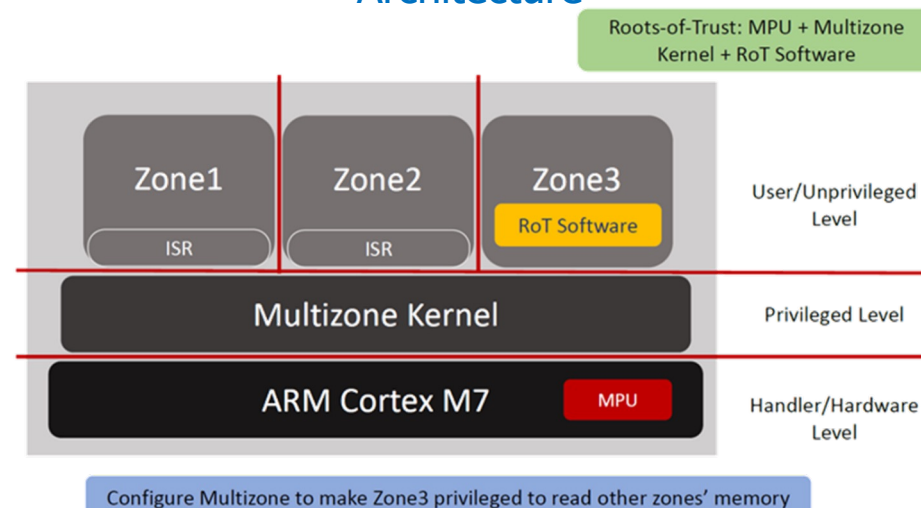
Peraton Labs: Software Root of Trust (RoT)

- Goal: design a RoT for TEE-less mid-tier devices
- Mid-tier: low cost, low power, single core, MPU, crypto hardware support, secure boot, no virtualization/MMU
- Challenges:
 - Key confidentiality:
 - Can store in read-only privileged memory region protected by MMU, but can be accessed by ISRs
 - Safely execute the RoT:
 - RoT code can be placed in privileged memory protected by MMU
 - Un-interruptability can be achieved by disabling hardware interrupt handler
 - Use supervisor call for controlled invocation of RoT software
 - *BUT* bugs in ISR can be exploited to modify MPU configuration
- Approach: move all user-dependent programs like ISRs to unprivileged level, keep only RoT and minimal bootloader code to privileged level
 - Multizone from HexFive Security:
 - Software-only alternative to TrustZone-M
 - Uses MPU to provide additional isolation guarantees

ARM Cortex M7



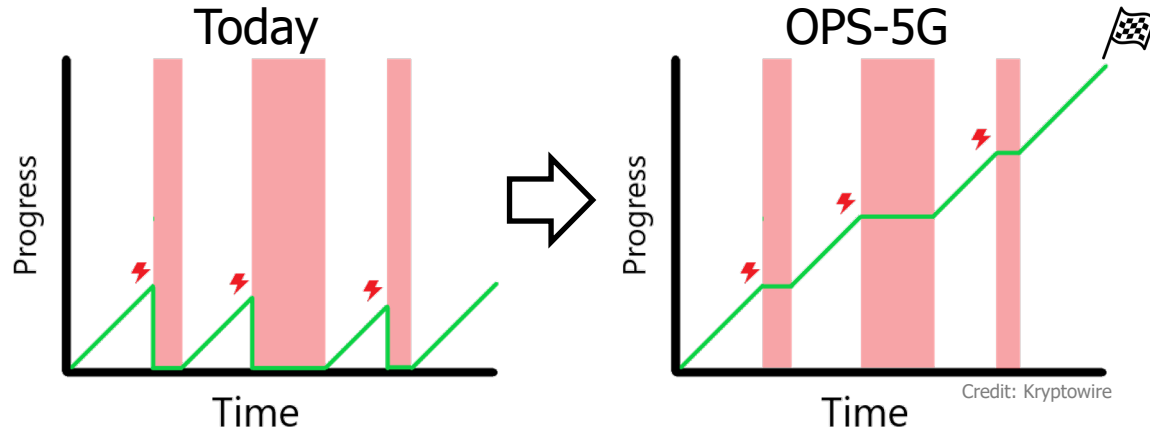
Architecture



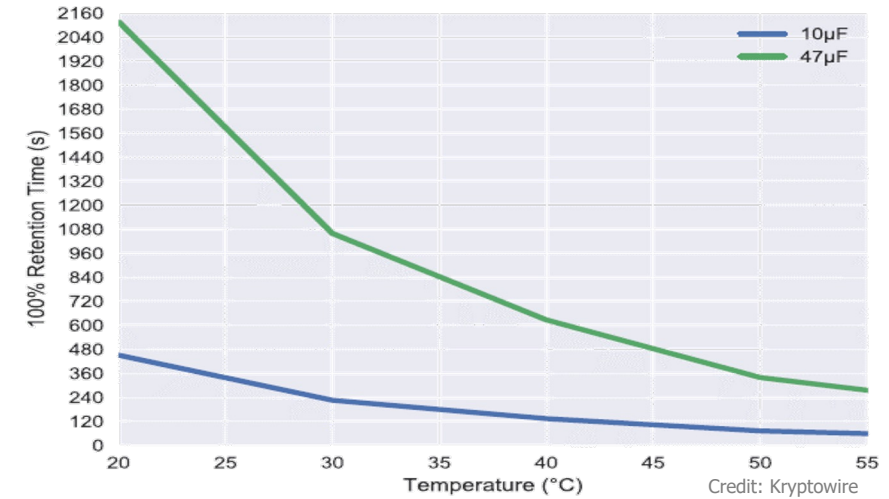


Kryptowire: intermittent computing and Time-Dependent Non-Volatility (TDNV)

Goal: Long-running computation through intermittent execution

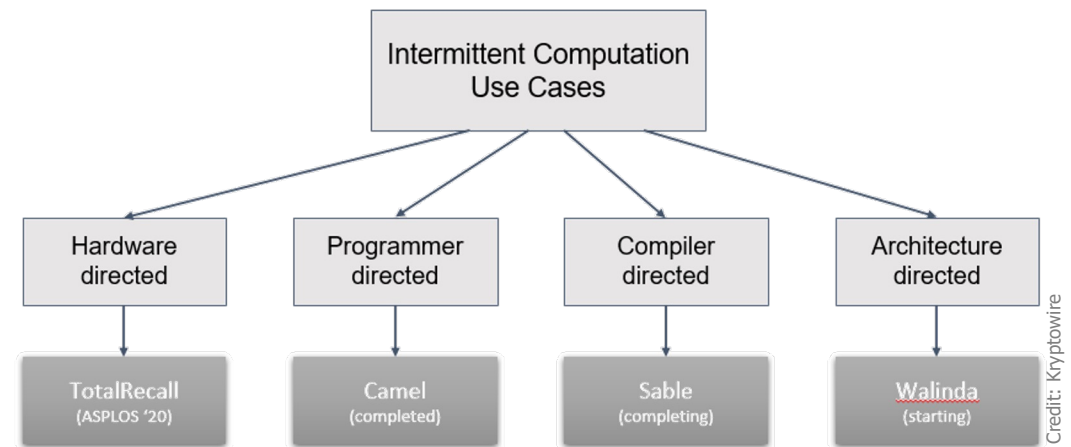
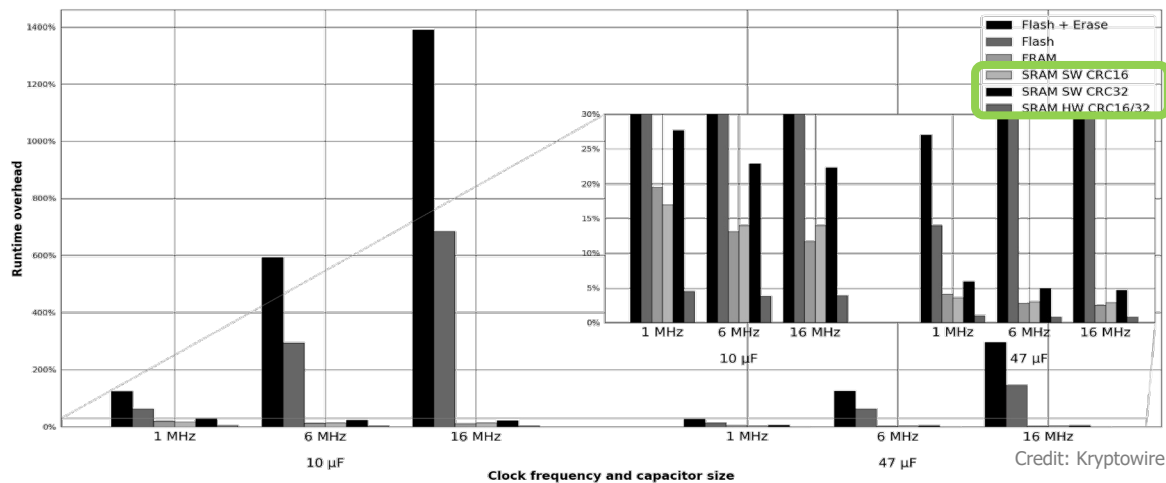


SRAM TDNV: state stability vs. temperature



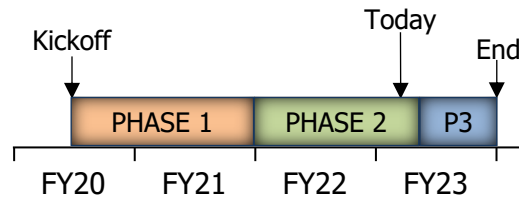
The time before the first SRAM cell in a microcontroller loses state varies with ambient temperature and capacitor size

Overhead of SRAM vs. flash-based checkpointing on CRC application



Williams et al. Forget Failure: Exploiting SRAM Data Remanence for Low-overhead Intermittent Computation. In *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS '20)*. Association for Computing Machinery, New York, NY, USA, 69–84.

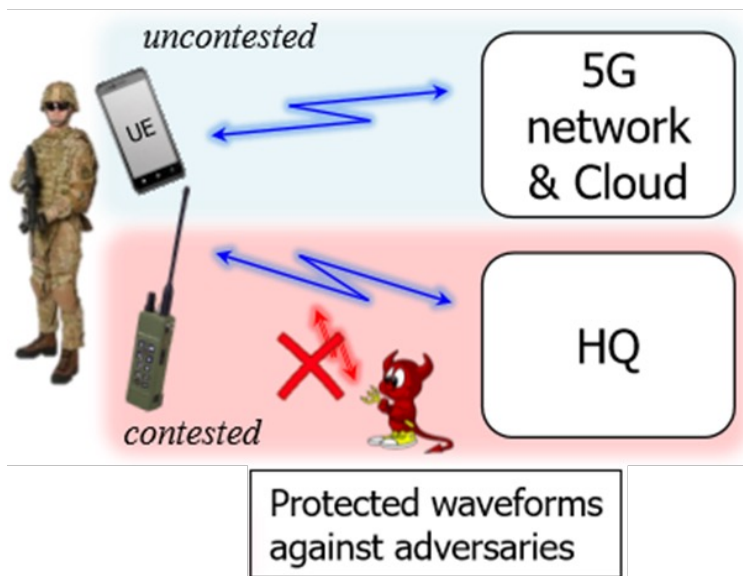
Open Programmable Accelerated 5G (OPA-5G)



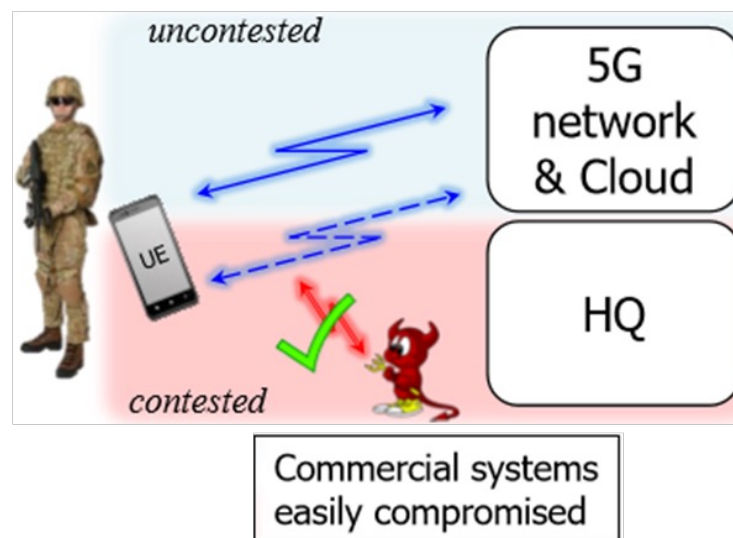
Acceleration of open RAN/core software stack to support 5G Super Blueprints (end-to end edge-core-RAN) for government, telecom, and enterprise end users

- Acceleration and innovation requires openness for 5G
- Improve the efficiency and security of deployments and operations
- Enable a more competitive, vibrant, interoperable, and trusted supplier ecosystem

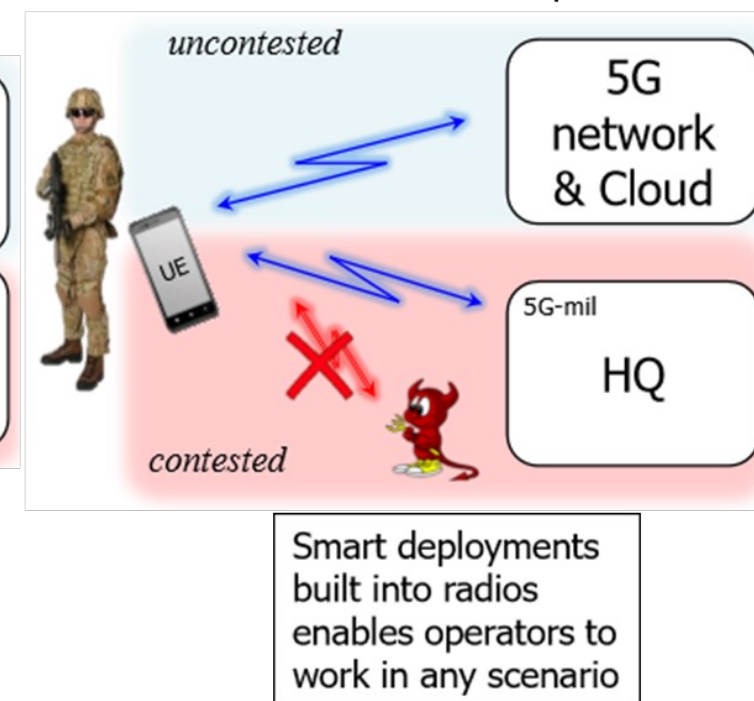
Ideal use of mobile phones



Real use of mobile phones



OPA-5G use of mobile phones





Vulnerabilities in

OPA-5G Plan

Standards

Weakness inherent to the 3GPP 5G standards releases



Open source hardware and software
Repeatability and transparency

Network

Weaknesses in network configuration and parameters



Full 5G UE and gNodeB configuration managements
Tied to OPS-5G program

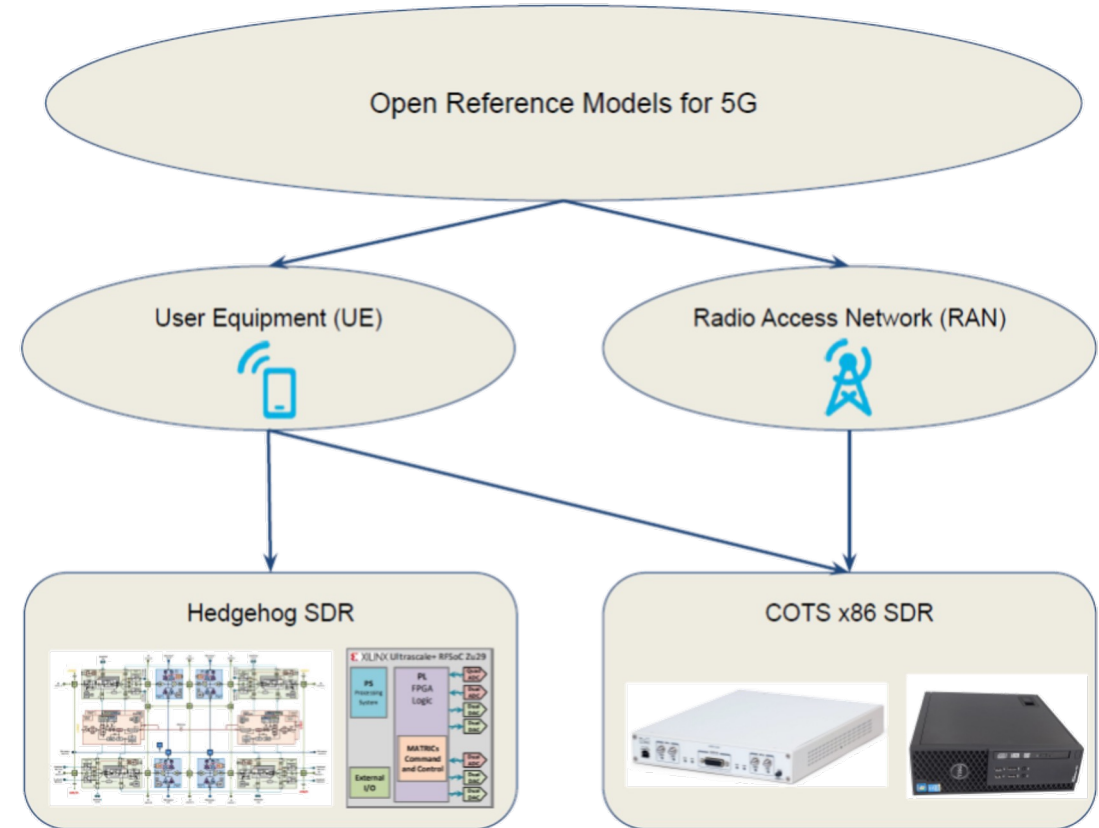
Implementation

Weaknesses in hardware and commercial software



Cooperation with community
DARPA, DoD labs, industry partners

OSS Design Flow



UE and gNB for 5G NSA and SA implementation released under DARPA OPA-5G



How is OPA contributing to open, secure, trusted 5G?

- Funding fully open source NSA and SA 5G software for the RAN to include uE and gNB
- OPA-5G government team porting code to hardware testbeds (x86 and FPGA) for testing and IOC demonstration
- Building diagnostic tools into software to provide real-time feedback and enable optimization
- Establishing hardware testbed to support 3GPP conformance testing for increased compliance to standards
- Establishing a DoD information repository for 5G RAN software baseline code for broader DoD adoption

Example of spectrogram of 5G channel captured using OPA-5G code

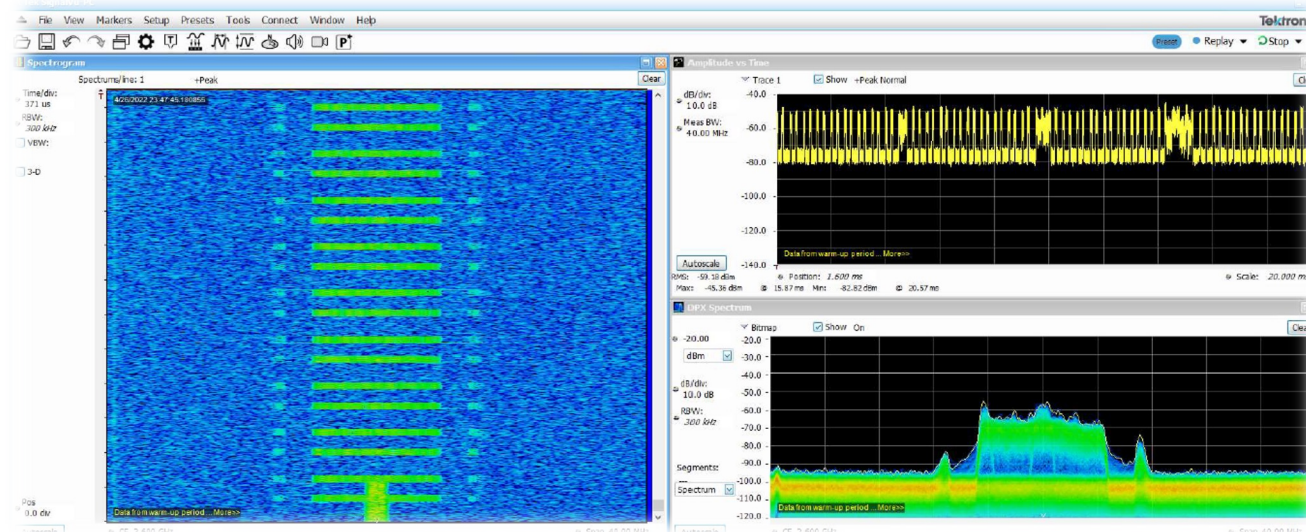
```
user@Puget-204474-1: ~/sonic-sw-next-sauE/usr/src
$ ./stoping ...
... exiting ...
$ sudo ./configure --prefix=/usr --sonic-sw-next-sauE/usr/src $ sudo srsvs ue.conf
Active RF plugins: librsran_rf_uhd.so librsran_rf_zmq.so
Inactive RF plugins:
Loading configuration file ue.conf...
WARNING: cpu0 scaling governor is not set to performance mode. Realtime processing could be compromised. Consider setting it to
performance mode before running the application.

Built in Release mode using 22.04.0.

Opening 1 channels in RF device=uhd with args=type=x300,serial=319D7EB,clock=external,sampling_rate=23.04e6,lo_freq_offset_hz=
None,None
Supported RF device list: UHD zmq file
INFO] [UHD] linux; GNU C++ version 9.3.0; Boost 107100; UHD 3.15.0-HEAD-0-gaae0e2de
INFO] [LOGGING] Fastpath logging disabled at runtime.
Opening USRP channels=1, args= type=x300,serial=319D7EB,lo_freq_offset_hz=23.04e6,None,naster_clock_rate=184.32e6
INFO] [X300] RF use specific instance constructed
INFO] [X300] X300 initialization sequence...
INFO] [X300] Maximum frame size: 8000 bytes.
INFO] [X300] Radio 1x clock: 184.32 MHz
INFO] [0/DmaIF0_0] Initializing block control (NOC ID: 0xf1f0d00000000000)
INFO] [0/DmaIF0_0] BIST passed (Throughput: 1293 MB/s)
INFO] [0/DmaIF0_0] BIST passed (Throughput: 1302 MB/s)
INFO] [0/Radio_0] Initializing block control (NOC ID: 0x12AD100000000001)
INFO] [0/Radio_1] Initializing block control (NOC ID: 0x12AD100000000001)
INFO] [0/DDC_0] Initializing block control (NOC ID: 0x0DC0000000000000)
INFO] [0/DDC_1] Initializing block control (NOC ID: 0x0DC0000000000000)
INFO] [0/DDC_2] Initializing block control (NOC ID: 0x0DC0000000000000)
INFO] [0/DDC_3] Initializing block control (NOC ID: 0x0DC0000000000000)
INFO] [0/BUC_0] Initializing block control (NOC ID: 0x0DC0000000000000)
INFO] [0/BUC_1] Initializing block control (NOC ID: 0x0DC0000000000000)
Waiting PHY to initialize ... done!

$ ./runchm UE
Random Access Transmission: prach_preamble=0, preamble_index=0, ra_rnti=0xf, tti=4971
Random Access Complete. c-rnti=0x400f, ta=2
RCNR Connected
RCNR reconfiguration successful.
RRC Session Establishment successful. IP: 192.168.3.2
RCNR reconfiguration successful.
$ ./stoping ...
```

Credit: SRS



Credit: SRS

- Completed successful SA attach with commercial tool
- Successful NR acquisition, session establishment

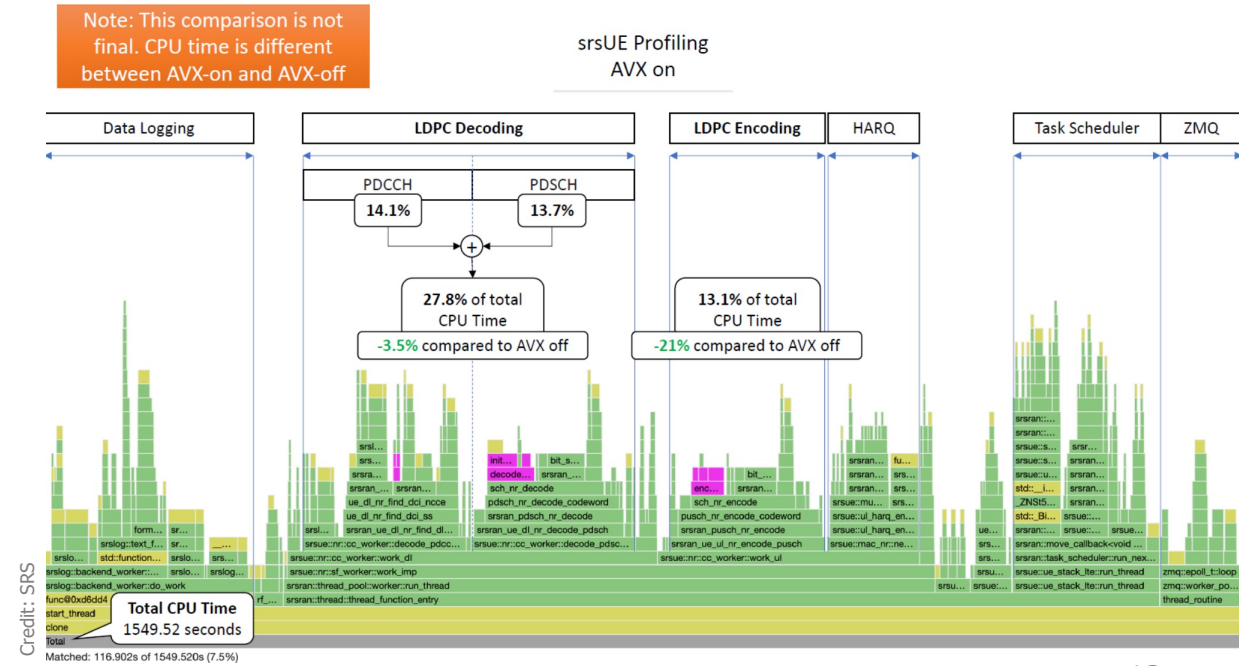
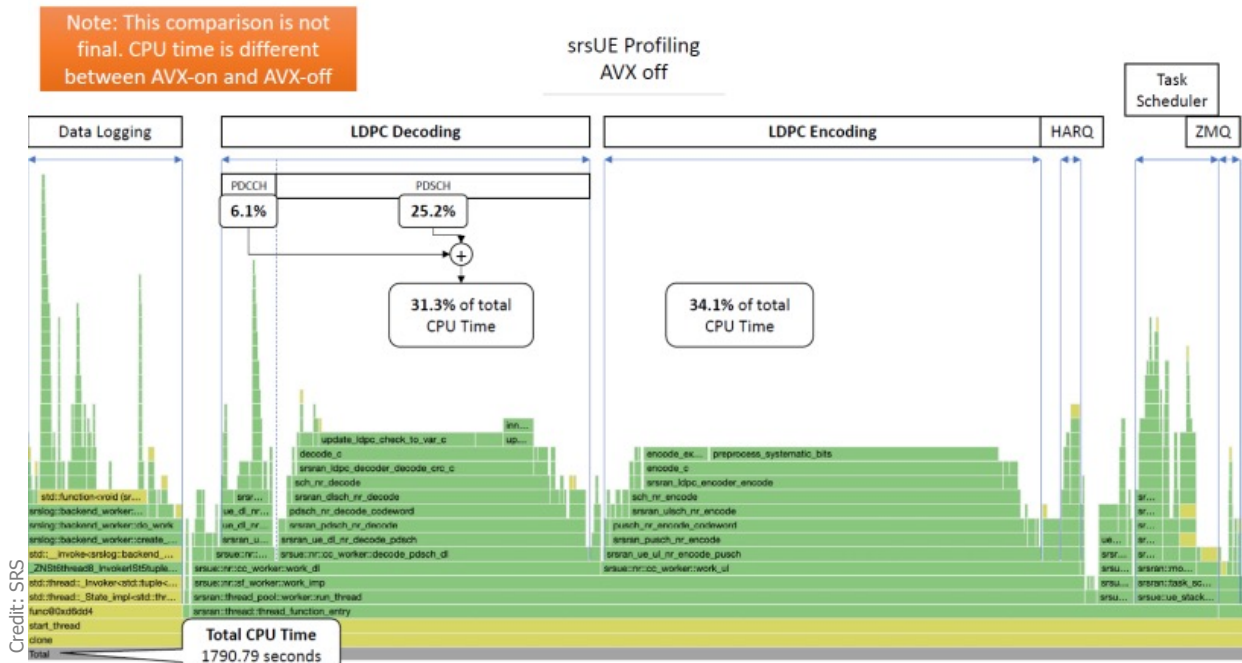
- Identified issue in code and parameter configurability needed by DoD



Open source code stack enables introspection and insight

- OPA-5G conducted profiling of the code base:
 - Intel Vtune: generates profiling data and visualizes results
 - Gprof: generates profiling data
 - Perf: generates profiling data
 - Flamegraph: visualization for profiling results
 - Valgrind/callgrind: generates profiling data
 - Frida-trace: dynamic function tracing
- Profiling of open and accessible code enables finer grained analysis and configuration optimization

Illustrative profiling of SRS code for different accelerator configurations





5G DoD Information Repository and Reference Implementation Lab

Capturing DoD investment in open, secure, and trusted 5G

The Information Repository (IR) is:

- The authoritative source for DoD waveforms and associated communications products
- Contains waveform software source code and supporting technical data
- A cyber-hardened information system developed by the Joint Tactical Networking Center in accordance with DoD Instruction 4630.09

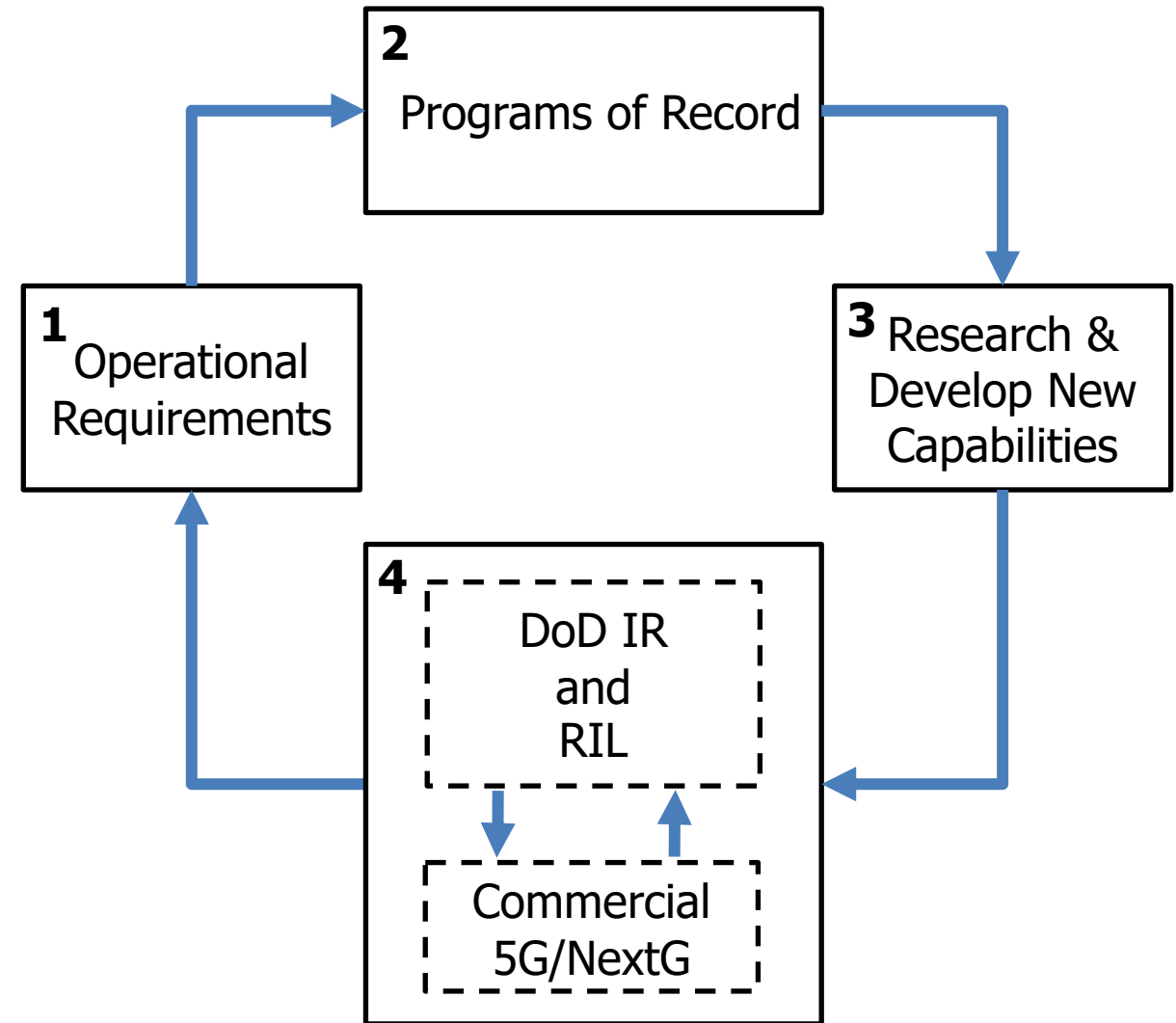
The Reference Implementation Lab (RIL) is:

- An implementation used as a definitive interpretation of the standards
- Utilized to mature, assess, integrate, and prototype innovations
- The culmination of processes resulting in proper operation in its communications waveform environment (to include instantiation, configuration, customization, testing, and systems integration)



OPA-5G: open software to accelerate DoD research and development in 5G

1. End user assessments to identify needs and capability gaps
2. Stakeholders engagement
3. Open, trusted, secure waveform research and development
4. DoD IR captures government investments and publishes waveform; RIL conducts integration and test





www.darpa.mil