

Secure FPGA Multi-Tenancy in the Cloud: Challenges and Opportunities

Ghada Dessouky, Ahmad-Reza Sadeghi and Shaza Zeitouni

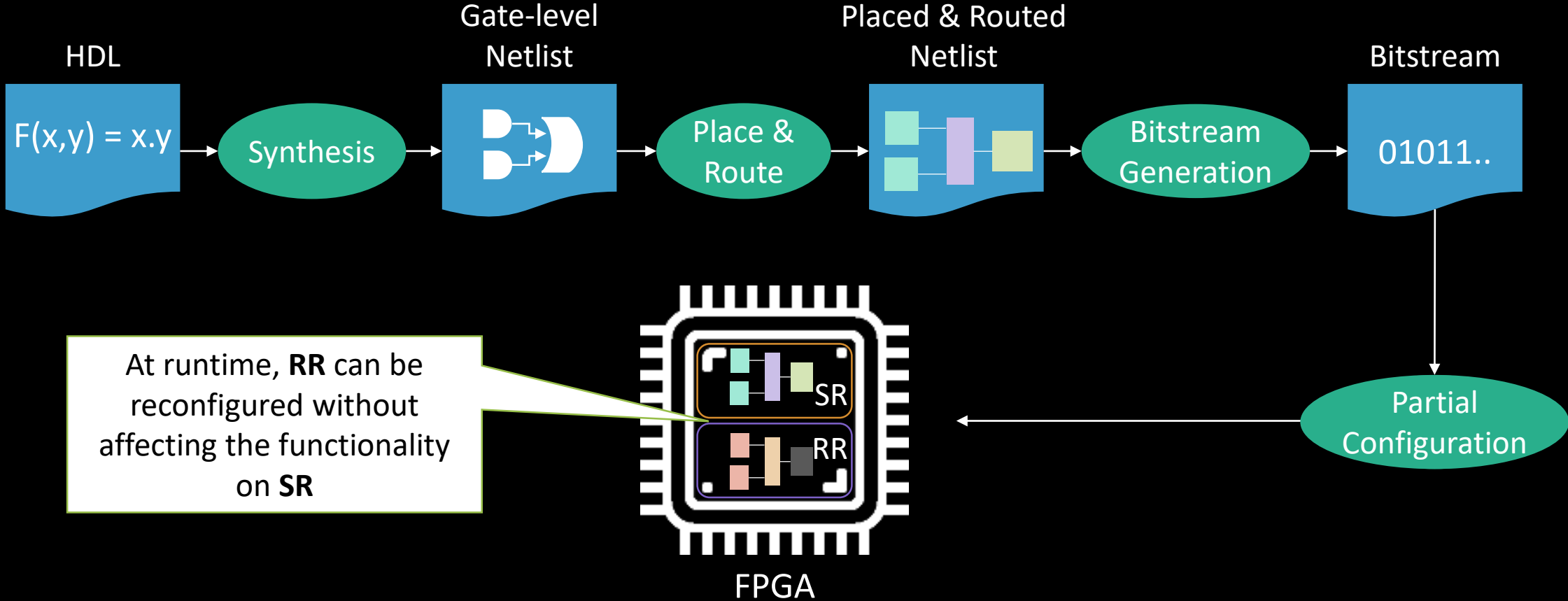
Technische Universität Darmstadt, Germany


{ghada.dessouky, ahmad.sadeghi, shaza.zeitouni}@trust.tu-darmstadt.de

Outline

- FPGA Workflow
- Deployment Models
- Adversary Model
 - Untrusted Clients
 - Traditional Attacks & Defenses
 - Remote Physical Attacks & Defenses
 - Untrusted Cloud Provider/Operator
 - IP Protection
- Challenges of Trusted Cloud FPGA Computing
- Future Research Directions

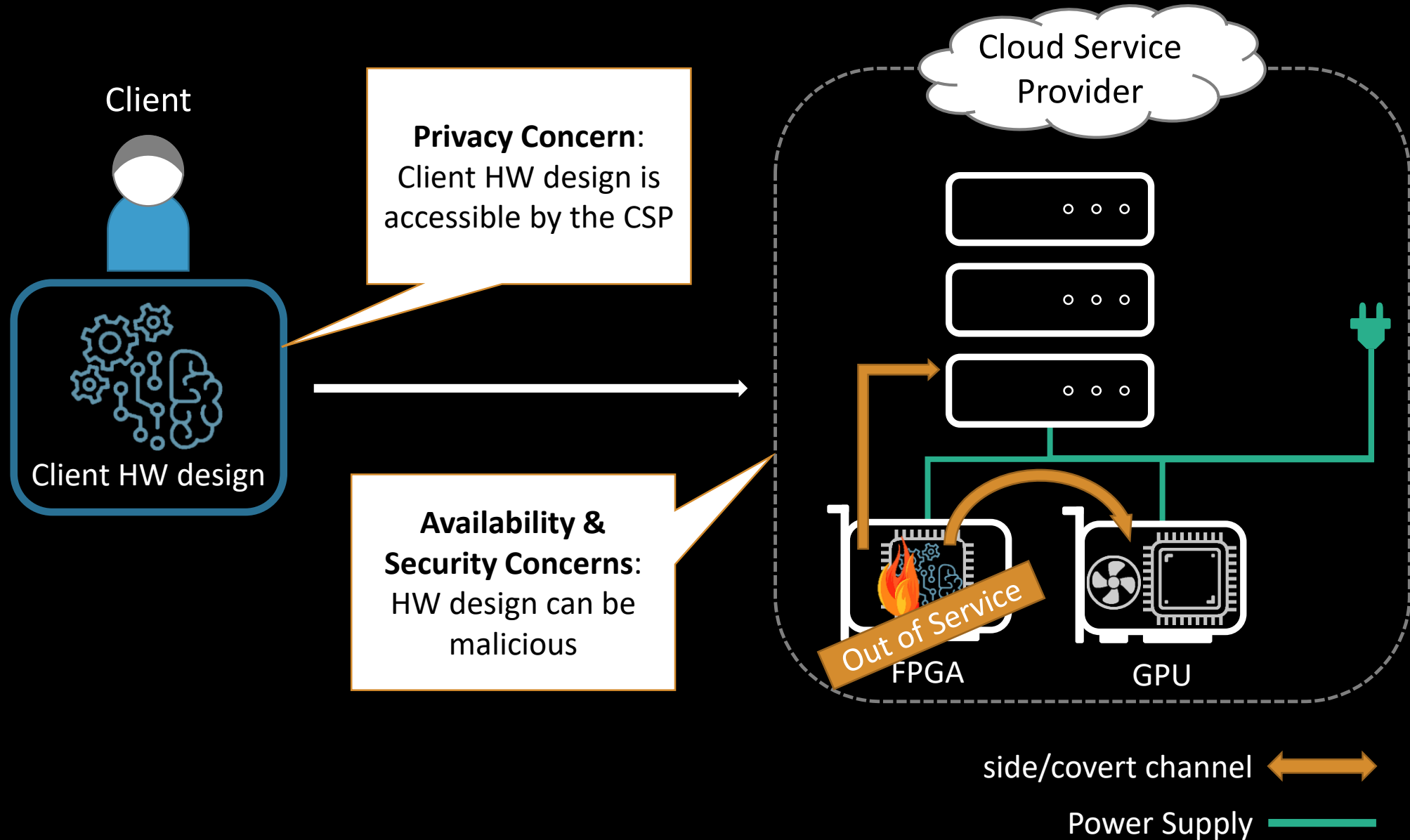
Background: FPGA Design Flow



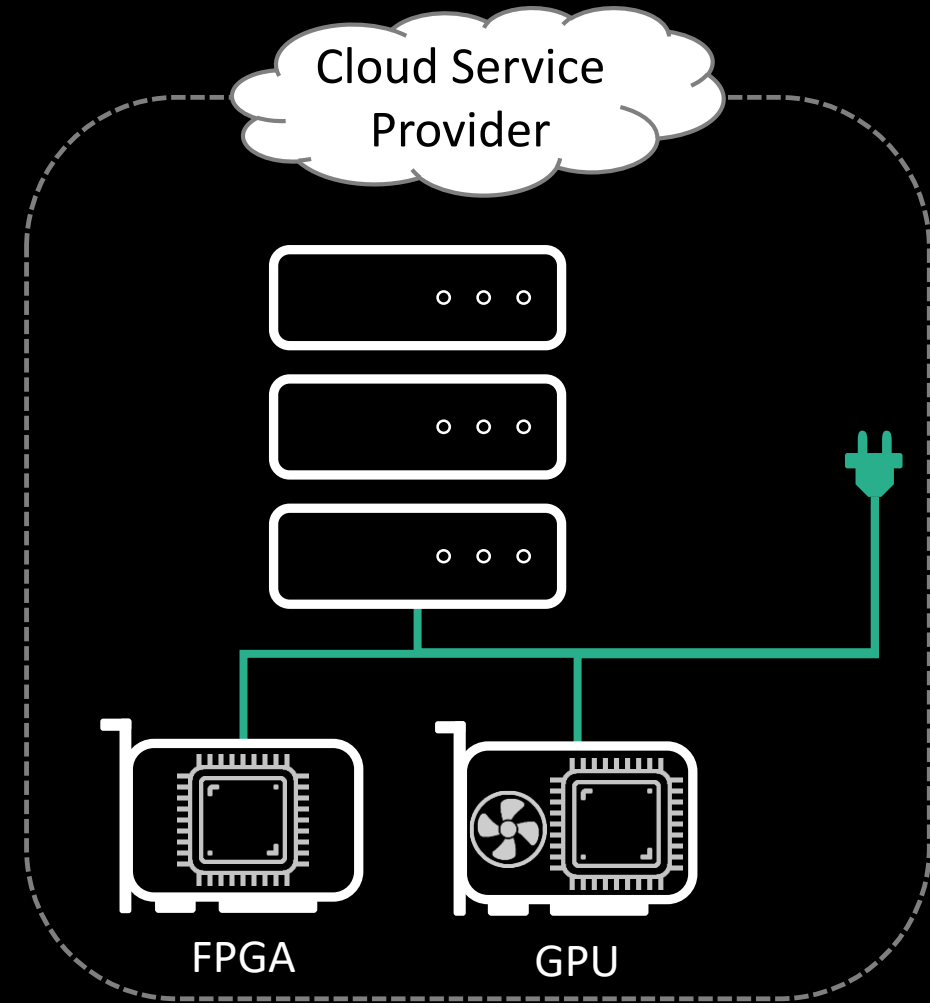
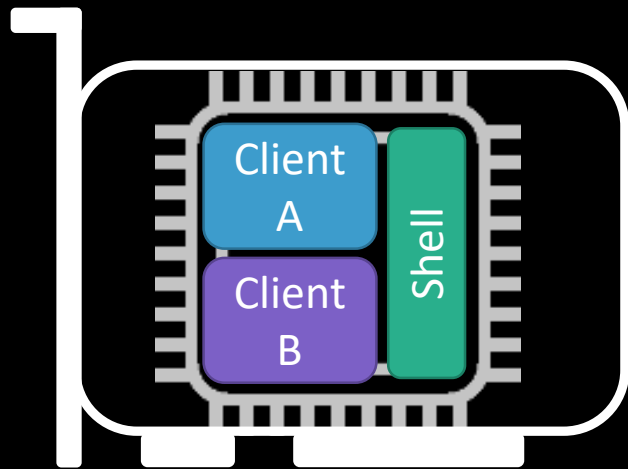
Process 

SR: static region RR: reconfigurable region

FPGA as a Service (FaaS)

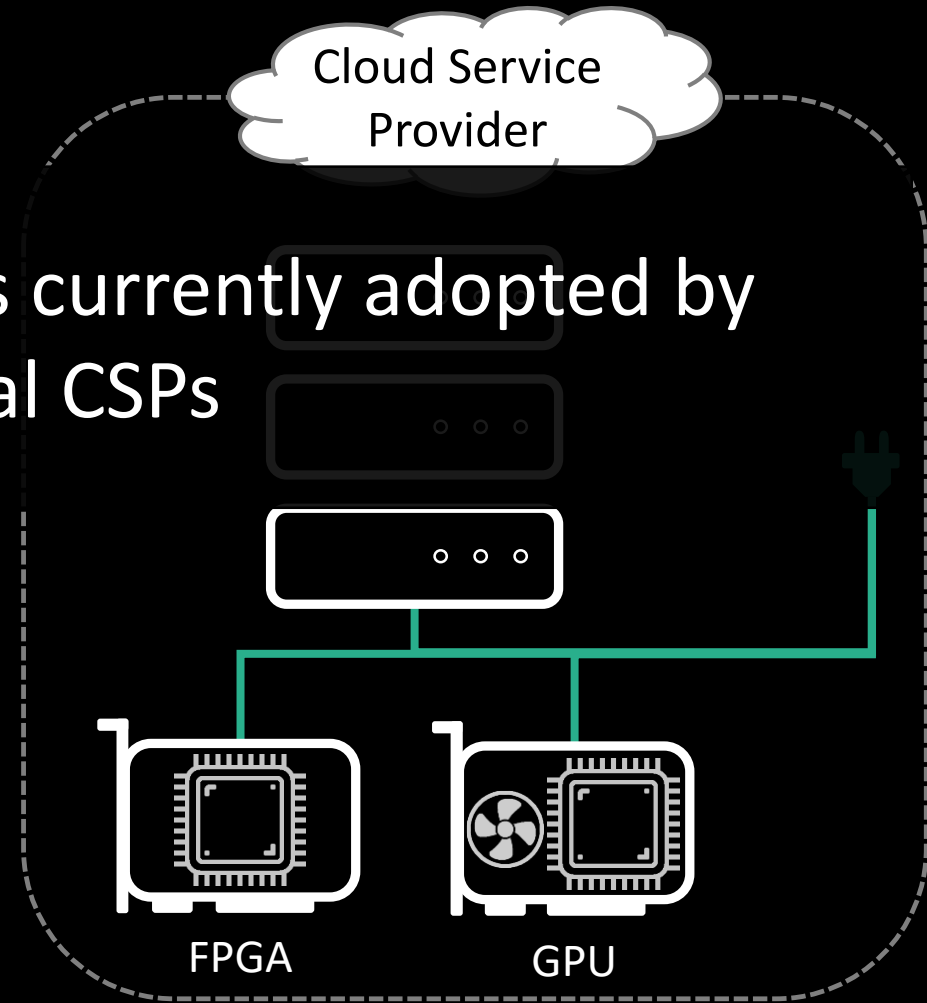
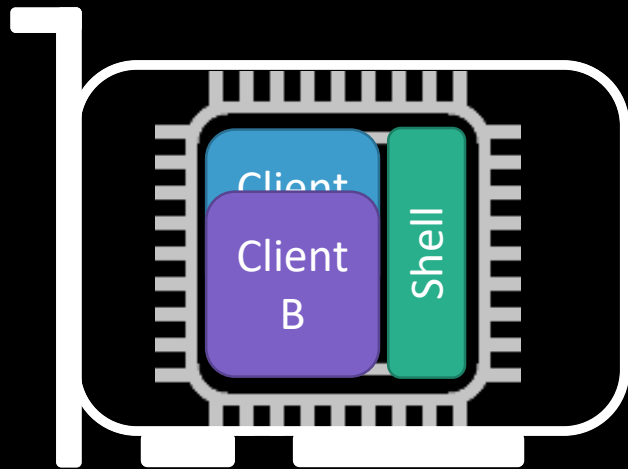


FaaS: Spatial Multi-Tenancy



FaaS: Temporal Multi-Tenancy

Temporal Multi-Tenancy is currently adopted by commercial CSPs



Adversary Model

➤ *Malicious clients*

- Traditional Attacks: e.g. Rowhammer and cache side-channel attacks
- FPGA-specific Attacks: due to rogue FPGA hardware designs

➤ *Untrusted cloud provider/operator*

- IP Piracy: FPGA configuration is not allowed to be encrypted

Traditional Attacks

Variants of CPU-based computing attacks

FPGA Accelerators with Shared Caches*

- The FPGA has a dedicated local cache
 - Coherent with the CPU's cache
- FPGA accelerator & CPU cores share the last level cache (LLC) and the main memory
 - Reading from main memory by the FPGA **does not affect** the shared LLC, but **writing does!**
 - No clflush!

FPGA Rowhammer & Cache Attacks

Rowhammer Attacks [74]

- No flushing is required
- Compared to CPU: faster memory accesses -> more bit flips

Cache Attacks [74]

Attacker-Victim	CPU-FPGA	FPGA-CPU
Flush-based	✓	✗
Eviction-based	✗	✓

Same defenses as in CPU-based Computing Paradigm!

Remote Physical Attacks

Due to **rogue** FPGA hardware designs (configurable sensors and power viruses)

Remote Physical Attacks in Spatial Settings

Target/Victim	FPGA (region-to-region)		CPU*
Effect	Power	Crosstalk	Power
DoS	[14]		
Side-channel	[15-18]	[23-27]	[18,19]
Covert-channel	[21,22]	[25,26]	

* In FPGA-SoC platforms, e.g., Xilinx Zynq boards

Remote Physical Attacks in Temporal Settings

Target/Victim	FPGA-to-FPGA		CPU	GPU
Effect	Power	Temperature	Power	Power
DoS	[11,12]			
Side-channel	[28]			
Covert-channel	[29]	[30]	[29]	[29]

Remote Physical Attacks

- These attacks are conducted by unprivileged malicious client
 - Exploit the FPGA fabric (configure with malicious circuits)
 - Can be also launched by the privileged FPGA shell
- Similar to Power Drop & Power Leakage Attacks on CPUs (TEEs) [110-112]
 - Exploit software-accessible interfaces to dynamic voltage & frequency scaling

Defenses Against Remote Physical Attacks

Runtime defenses

- May use rogue primitives, e.g., sensors
- High Overhead (area & power)
- Detecting/hiding **individual** effects (thermal/power changes)
- Should be tuned to specific FPGA and protected hardware design

Preventive defenses: Virus Scanners

- Searching hardware designs for **known** rogue primitives (power viruses & sensors)
- Should be updated to detect new rogue primitives
- Need access to **plain-text** FPGA configuration

IP Protection in Cloud FPGAs

Defending against Untrusted Cloud Providers/Operators

IP Protection against IP Piracy

- Existing solutions extend the FPGA shell to
 - Exchange secret key with the client
 - Decrypt client bitstream
 - Partially configure client application
- Require FPGA vendor support
- Work for temporal multi-tenancy
- **Encrypted** FPGA bitstreams may contain **rogue** primitives

Trusted Computing on Cloud FPGAs: Challenges

- Enabling trusted **configuration & execution** on cloud FPGAs
 - Minimal changes to FPGA fabric
 - FPGA shell/hypervisor is untrusted
 - Require vendor support
- Protecting cloud assets and co-clients by preventing malicious FPGA configurations
 - May require access to plaintext FPGA configurations
 - Enforced by FPGA vendors/toolchains → reduce flexibility

Future Directions

- Investigating FPGA Trusted Computing Base (FPGA-TCB)
 - Which components in software/built-in/configurable
- Hardening cloud FPGAs against remote physical attacks
 - Untrusted FPGA shell
 - Untrusted clients