TECHNISCHE UNIVERSITÄT DRESDEN

Fakultät Informatik, Institut für Technische Informatik, Professur für VLSI-Entwurfssysteme, Diagnostik und Architektur

Virtualisierte rekonfigurierbare Ressourcen und deren sichere Bereitstellung in einer nicht vertrauenswürdigen Cloud Umgebung

Zwischenvortrag

Paul R. Genßler – paul.genssler@tu-dresden.de

Dresden, 21. September 2017

Ziele der Arbeit

- Literaturstudium relevanter kryptographischer Verfahren und bisheriger Arbeit zum Thema Authentifizierung eines Bitstreams
- Konzeptionierung eines Host/FPGA-Hypervisors zur Ver- und Entschlüsselung partieller Bitstreams
- Prototypische Umsetzung des Konzepts auf Basis des RC2F
- Messung relevanter Leistungsdaten

1. Motivation

2. Literatur

3. Design

4. Weitere Schritte und Zusammenfassung

1. Motivation

2. Literatur

3. Design

4. Weitere Schritte und Zusammenfassung

Field Programmable Gate Array

- Flexibel konfigurierbare Hardware
- Auslagerung von Anwendungen, Sicherheit durch spezialisierte Designs
- Gewinnen an Bedeutung im Data Center

- FPGAs in der Cloud
  - Amazon EC2 F1 Instances[1], Microsoft Catapult[2]
  - Steigerung der Energieeffizienz
  - Virtualisierung nötig
  - Erhöhung der Auslastung großer Chips

[1] https://aws.amazon.com/ec2/instance-types/f1/

[2] Caulfield u. a., 2016: "A cloud-scale acceleration architecture"

Sicherheitsbedenken sind größtes Hemmnis

- Verlust der Herrschaft über die Daten (58 %[3])
- Kein Schutz des geistigen Eigentums
- Vielschichtige Umgebung → Viele Einfallstore
- Kleinere Angriffsfläche von Hardware nutzen

[3] Heidkamp u. a., 2016: *Cloud-Monitor 2016*

2 Literatur

Virtualisierung von FPGAs

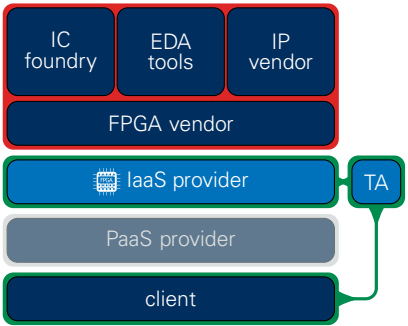
Byma u. a.	Virtualisierung mit OpenStack [4]
Fahmy u. a.	Erweiterung um Prioritäten und Inter-VM Kommunikation [5]
Asiatici u. a.	High-Level APIs und Virtualisierung des Speichers als HW/SW Schnittstelle [6]
Knodel u.a.	Flexible Partitionierung mit Service Modellen [7]

[4] Byma u. a., 2014: "FPGAs in the Cloud: Booting Virtualized Hardware Accelerators with OpenStack"
[5] Fahmy u. a., 2015: "Virtualized FPGA Accelerators for Efficient Cloud Computing"
[6] Asiatici u. a., 2016: "Designing a virtual runtime for FPGA accelerators in the cloud"
[7] Knodel u.a., 2017: "Virtualizing Reconfigurable Hardware to Provide Scalability in Cloud Architectures"

Sicherheitskonzepte

- Trusted Platform Model (TPM)
- Homomorphe Verschlüsselung

- FPGA Sicherheitskonzept vorhanden
- Einbeziehung einer Trusted Authority (TA)
  - Kepa u. a., 2008
  - Devic u. a., 2010
  - Eguro u. a., 2012

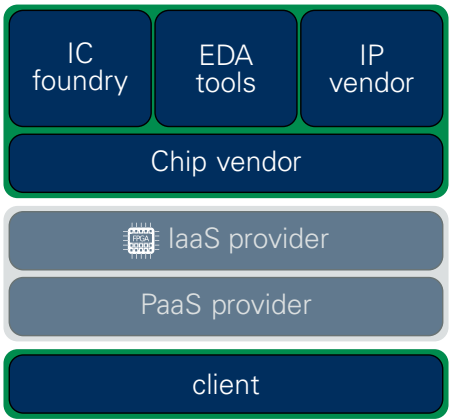


Gefahrenmodell

- \- Unsicher nach Versand durch Hersteller
- \- Trägerboard und externe Komponenten
- \- Cloud Software
- \+ Algorithmen sicher
- \+ Hardware abgesichert
- \+ Lokale Software vertrauenswürdig
- • Keine DoS-artigen Angriffe

3 Design

Ziel des Entwurfs



3 Design

Motivation

Cloudanbieter (IaaS)

- Ressourcenverwalter
- Flexibilität, Skalierbarkeit, schnelle Bereitstellung
- Geringer CPU Overhead

Mittelständige Firma

- Auslagerung der Datenverarbeitung in die Cloud
- Sensible Daten → keine Public Cloud
- Hoher Durchsatz, viel Speicher
- Sicherer, authentifizierbarer Endpunkt

3 Design

TLS

- SSL 1.0 - 3.0, TLS 1.0 - 1.2
- Standard für sichere Kommunikation im Internet
- Viele Verschiedene Algorithmen (Cipher Suites)

Asymmetrische Verschlüsselung

- RSA, ECIES
- Privater Schlüssel: Entschlüsselung
- Öffentlicher Schlüssel: Verschlüsselung
- Basiert auf mathematischen Problemen ohne effiziente Lösung
- Schlüsselgrößen: 2048 Bit (RSA), 256 Bit (ECIES)

3 Design

Digitale Signatur

- RSA, ECDSA
- Öffentlicher Schlüssel: Verifizierung
- Privater Schlüssel: Signierung
- Basiert auf gleichen Prinzipien wie asymmetrische Verschlüsselung
- Schlüsselgrößen: 2048 Bit (RSA), 256 Bit (ECDSA)

3 Design

TLS

- SSL 1.0 - 3.0, TLS 1.0 - 1.2
- Standard für sichere Kommunikation im Internet
- Viele Verschiedene Algorithmen (Cipher Suites)

Symmetrische Verschlüsselung

- AES, DES, KASUMI
- RC5, Chacha20
- Gleicher Schlüssel zum Ver- und Entschlüsseln
- Konfusion und Diffusion [8]
- Schlüsselgrößen: 128 - 256 Bit

[8] Shannon, 1945: "A Mathematical Theory of Cryptography"

3 Design

TLS

- SSL 1.0 - 3.0, TLS 1.0 - 1.2
- Standard für sichere Kommunikation im Internet
- Viele Verschiedene Algorithmen (Cipher Suites)

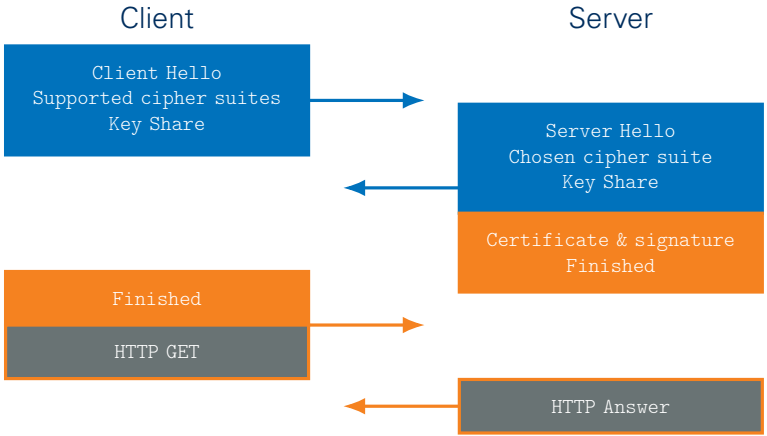
Hashfunktionen

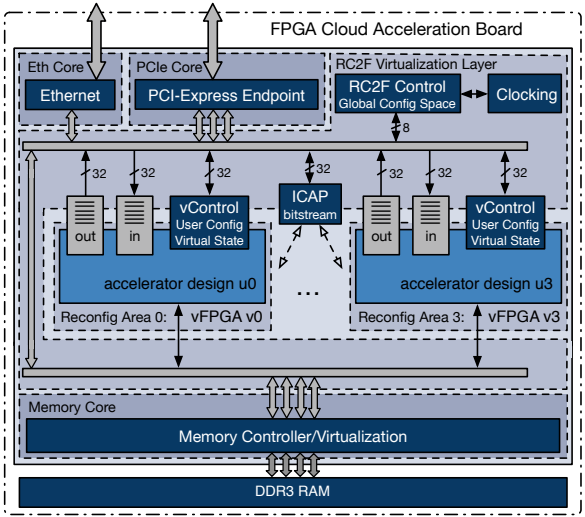
$$h(\text{📄}) = 0x4914D65BC6140743C5B6719A79746DB1_{16}$$

- MD5, SHA1, SHA224, SHA256, SHA3-512
- Gleichverteilte Abbildung beliebig langer Daten auf Hash fester Länge
- Gleiche Eingabe erzeugt gleiche Ausgabe
- Konfusion und Diffusion
- Kollisionsresistenz
- Rekonstruktion der Eingabe praktisch unmöglich

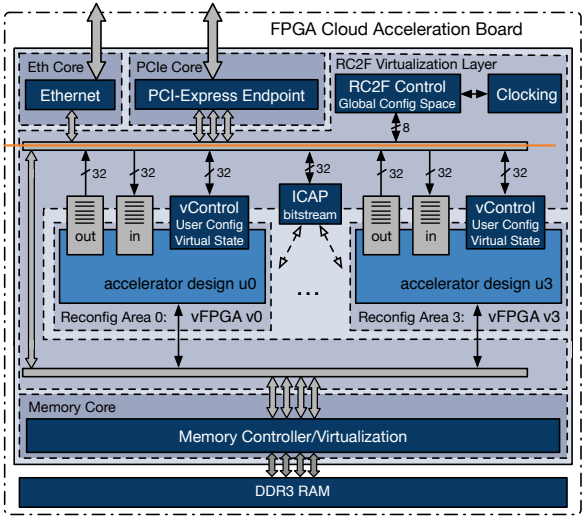
3 Design

TLS 1.3 Handshake

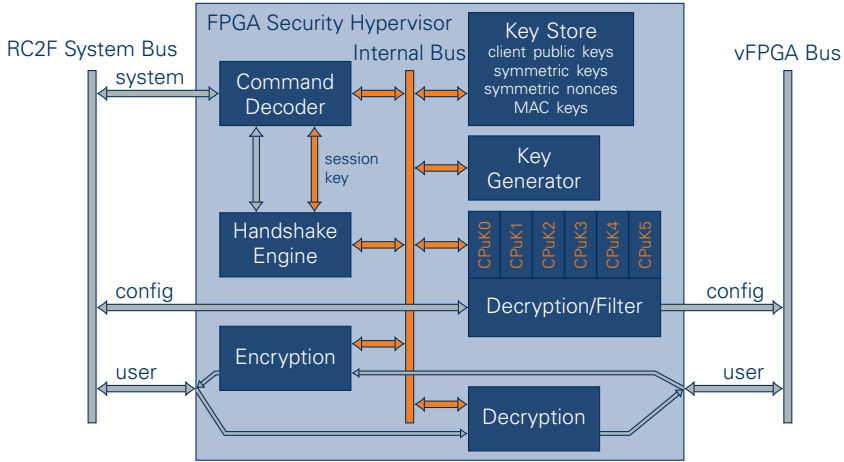




Knodel u. a., 2016 "RC3E: Reconfigurable Accelerators in Data Centres and their Provision by Adapted Service Models"



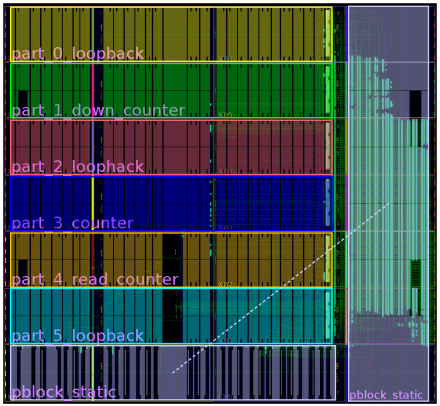
Knodel u. a., 2016 "RC3E: Reconfigurable Accelerators in Data Centres and their Provision by Adapted Service Models"


FPGA Security Hypervisor
RC2F System Bus
Internal Bus
vFPGA Bus
system
Command Decoder
session key
Handshake Engine
Key Store
client public keys
symmetric keys
symmetric nonces
MAC keys
Key Generator
CPuK0
CPuK1
CPuK2
CPuK3
CPuK4
CPuK5
config
Decryption/Filter
config
Encryption
user
user
Decryption

3 Design

Partielle Rekonfiguration

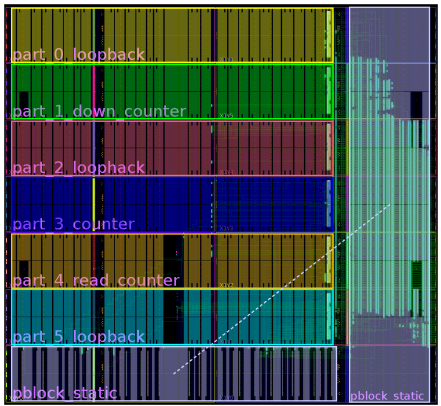
- 6 nutzbare Regionen
- Beeinflussung eines anderen Nutzers
- Format der Konfigurationsdatei basiert auf Frames
- Entschlüsselung und Analyse in Hardware



3 Design

Rekonfigurationsfilter

- Digitale Signatur in Konfigurationsdatei
- 5 Adressbereiche
- Filter für unerlaubte Bereiche
- Gemeinsame Konfiguration von speziellem Bereich
- Einfluss an Grenzen
- Tiefgreifende Analyse schwierig



1. Motivation

2. Literatur

3. Design

4. Weitere Schritte und Zusammenfassung

Weitere Schritte und Zusammenfassung

- Weitere Schritte
  - Implementierung abschließen
  - Leistungsdaten messen
  - Notwendige Architekturänderungen aufzeigen
- Zusammenfassung
  - Absolute Sicherheit nicht möglich
  - Verbesserung durch Einsatz teilweise rekonfigurierbarer Hardware
  - Nutzung vorhandener Protokolle und Algorithmen

Devic, Florian u. a. (2010). “Secure protocol implementation for remote bitstream update preventing replay attacks on FPGA”. In: *Field Programmable Logic and Applications (FPL), 2010 International Conference on*. IEEE, S. 179–182.

Eguro, K. u. a. (2012). “FPGAs for trusted cloud computing”. In: *22nd International Conference on Field Programmable Logic and Applications (FPL)*, S. 63–70. DOI: 10.1109/FPL.2012.6339242.

Kepa, Krzysztof u. a. (2008). “Serecon: A secure dynamic partial reconfiguration controller”. In: *Symposium on VLSI, 2008. ISVLSI’08. IEEE Computer Society Annual*. IEEE, S. 292–297.