Telefónica

Serverless Computing: A Security Perspective

Dr. Eduard Marin

Telefonica Research

FLUIDOS

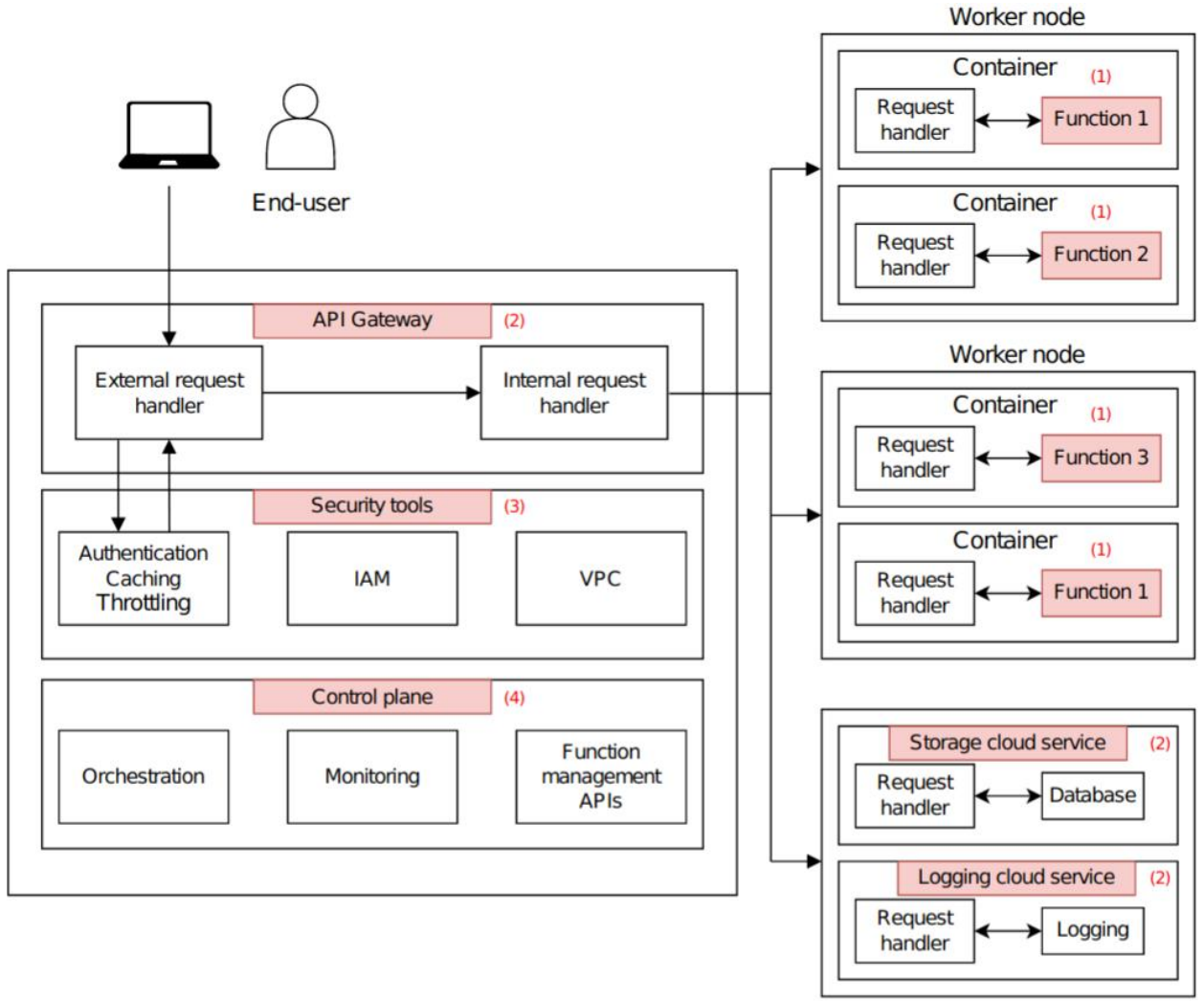
EDGELESS

nebulous

Serverless computing

- Key advantages:
  - All infrastructure management and operational tasks are outsourced to cloud providers
  - Pure pay-per-use
  - Simpler, more effective and less costly auto-scaling
- Backend-as-a-Service (BaaS) vs. Function-as-a-Service (FaaS)
- Compute and storage are decoupled from each other
  - Application logic: a set of small, short-lived and stateless functions
  - Cloud services offer storage (and more)
  - Event-based (e.g., HTTP requests, modification to objects in storage, etc)

FaaS serverless architecture (simplified)



Security of serverless computing

- Better or worse than its predecessors?
- Shared responsibility between cloud providers and app owners
- Serverless security requires a major change in mindset from software developers
- Need for strong security and privacy by design in all stages of a function
  - Function development
  - Function placement
  - Function execution

Serverless security

- **Functions are stateless, short-lived and single purpose**
- Security is a shared responsibility
- Mitigates DoS attacks

- Large attack surface
- Security vs performance vs cost
- Cloud provider backends

Serverless security

- Functions are stateless, short-lived and single purpose
- **Security is a shared responsibility**
- Mitigates DoS attacks

- Large attack surface
- Security vs performance vs cost
- Cloud provider backends

Serverless security

- Functions are stateless, short-lived and single purpose
- Security is a shared responsibility
- **Mitigates DoS attacks**

- Large attack surface
- Security vs performance vs cost
- Cloud provider backends

Serverless security

- Functions are stateless, short-lived and single purpose
- Security is a shared responsibility
- Mitigates DoS attacks

- **Large attack surface**
- Security vs performance vs cost
- Cloud provider backends

Attack surface

- Lots of function-to-function and function-to-cloud service communication
- Functions are triggered by many internal and external events (47 in AWS Lambda) with multiple formats and encoding
- New components and cloud services, many of which are shared across users (e.g., the storage service)

Serverless security

- Functions are stateless, short-lived and single purpose
- Security is a shared responsibility
- Mitigates DoS attacks

- Large attack surface
- **Security vs performance vs cost**
- Cloud provider backends

Security vs performance vs cost

- VMs vs containers vs custom execution environments
- Cold vs warm containers
- Deterministic vs random function placing algorithms

Serverless security

- Functions are stateless, short-lived and single purpose
- Security is a shared responsibility
- Mitigates DoS attacks

- Large attack surface
- Security vs performance vs cost
- **Cloud provider backends**

Cloud provider backends

- Cloud providers are largely responsible for providing security
- "False sense of security"
  - Software developers may ignore security in their applications
  - Or make unrealistic assumptions about the security mechanisms in place
- Information about backends is often kept confidential (or is not fully documented)

Attacks against serverless

- 1) Application-level attacks
- 2) Serverless-specific attacks
- 3) Hardware attacks (not covered in this talk)
  - Microarchitectural-type attacks (e.g., Meltdown, Spectre)
  - Rowhammer-type attacks
  - ....

Application-level attacks

1) Injection
2) Broken authentication
3) Sensitive data exposure
4) XML external entities
5) Broken access control
6) Security misconfiguration
7) Cross-site scripting
8) Insecure deserialization
9) Using components with known vulnerabilities
10) Insufficient logging and monitoring

Defenses against application-level attacks

- Treat every function as a security perimeter
- Adhere to standard secure coding best practices
- Follow the least privilege principle
- Secure data at rest and in transit (e.g., using TLS)

Serverless-specific attacks

- Resource exhaustion (e.g., denial-of-wallet)
- Attacks leveraging inconsistencies in functions and cloud services
- Side channel attacks
  - Based on access patterns or timing information
  - Intra-container and intra-host side channels
  - Attacks leveraging the disk space in /tmp

And many more!

Conclusions

- Security and privacy will be crucial for the widespread adoption of serverless computing

- "Serverless computing: a security perspective"
  - https://journalofcloudcomputing.springeropen.com/articles/10.1186/s13677-022-00347-w