Adapting ICN to Function Execution for Edge Computing
Function Execution in ICN

Any node can execute functions
Not just routing to the service
Natural load balancing
No need for DNS
Environment

- Requesting node
- Executing node
- Function provider
Function execution vs data retrieval

- Interests can trigger functions and retrieve data
- Execution time vary
- Execution is more costly than retrieving data
  - have to avoid double execution
  - easy DoS
Design inconsistencies
PIT Expiry

Delay is easy to predict with data retrieval

Functions add execution time that can vary

How long should a PIT entry remain in the table?
Pit Expiry

- Higher timer value
- Time in the Interest
- Persistent PIT
Mantaining a session

Multiple chunk result retrieval

Function interaction

We have no guarantee that consecutive interests will be received by the same node
Mantaining a session

- Unique node names/Callback
- Return data name only
- Indicate a path/Labels switching
Execute the function or look for cached results

• Finding cached data is also important in static content retrieval

• Cost of function execution is higher

• We want to execute the function as close to the source as possible...

• ... but we do not want to execute the function multiple times
Security
User identification

Signatures seem to be the best option...

...but require a key distribution scheme

Should the signature be a part of the name?
Signature

<table>
<thead>
<tr>
<th>Name</th>
<th>upIf</th>
<th>downIf</th>
</tr>
</thead>
</table>

PIT

![Diagram of a database with two people icons]
Packet spoofing
User authorization

How to determine if a user has access to a resource

Easy in a managed environment

Subscriptions

Border routers

Trusted environment
User Authorization

Difficult in unmanaged environment...

...but blockchain can help

Open, public ledger

High resource usage

What about the delay?
Communication encryption

User want to communicate with functions

The data should be invisible for the executing node/function provider
Intel SGX

Hardware based cryptography

Applications can form encrypted enclaves

An enclave is protected from the OS and cannot be accessed by anyone but the application

An enclave can be used to securely store a secret/private key
Intel SGX
Intel SGX - limitations

Application size limit (currently 90MB)

Need a TLS connection to transport a secret (only once)

.so .dll files only (extensions for docker being investigated)
Payments

- Untrusted environment
- Requesting node
  - does not know who will receive the request
  - have to make sure, the result is correct
- Executing node
  - does not want to work without a payment
Smart Contracts

Runs on blockchain

Publicly visible
Smart contracts
Thank you