Towards a GPU-Accelerated Domain Name System

Matthäus Wander, Johannes Brüderl

<matthaeus.wander@uni-due.de>

DNS and Internet Naming Research Directions
Marina del Rey, 2016-11-17
Motivation

- Cryptography is expensive for DNSSEC servers
  - NSEC3 iterated hashing
  - Online signing

Domain?

No.

NSEC3 proof:

\[ h_1 < h < h_2 \]

\[ h = \text{SHA1(SHA1(SHA1(domain)))} \]
Scalability

- How to scale up computing power?
  - Buy another server host
  - Buy a GPU: more computing power for less money

Domain?

No.
NSEC3 proof:
\[ h_1 < h < h_2 \]
\[ h = \text{SHA1}(\text{SHA1}(\text{SHA1}(\text{domain}))) \]
General-purpose Computing on GPUs

- Process batches of work by parallelization
- Significant speed-up of certain applications
  - e.g. NSEC3 zone enumeration (attacker side)
  - No branch divergence, few global memory accesses

\[ h = \text{SHA1(SHA1(SHA1(domain)))} \]
GPU Challenges

- GPU computing incurs **latency** penalty
  - Work batching
  - Move data between device boundaries
  - Lower clock frequency than CPU

- DNS operators are dead serious about latency
  - Trade-off: throughput vs. latency
Server Throughput under NSEC3 Load

DNS server throughput

Queries per second

Iterations

Nvidia GTX 970
AMD R9 390
Intel HD4000
Intel I5-3570k OpenCL
Intel I5-3570k iDNS
Response Latency

- Round-trip time with CPU
  - 0 iterations: < 1 ms per query
  - 150 iterations: beyond server capacity, drops queries

- Round-trip time with GPU and batching
  - 0 iterations: ~13 ms per query
  - 150 iterations: ~15 ms per query
  - 2500 iterations: ~45–50 ms per query
Outlook

- Strategy for GPU offloading?
- Batch sizing?
- Replace iterations with parallelizable hashing?
- Integrated Graphics Processors (IGPs)?
  - Zero-copy memory usage
- NSEC5 on GPU?