Chair of Network Architectures and Services
Department of Informatics
Technical University of Munich

Master's Thesis: Network Discovery Orchestration

Presenter: Michael Eder— *Advisors:* Simon Bauer, Jonas Jelten— *Supervisor:* Georg Carle

General Motivation

- Distributed port and higher layer network scans across an arbitrary number of hosts in a network
- Platform independent scanner nodes: runs without special privileges on any major OS and platform [2]
- Speed up scan (compared to nmap), generate views from different scan positions for your local/company network
- Think of Shodan [3]/Censys [1], with data generated from different scan positions for your local/company network

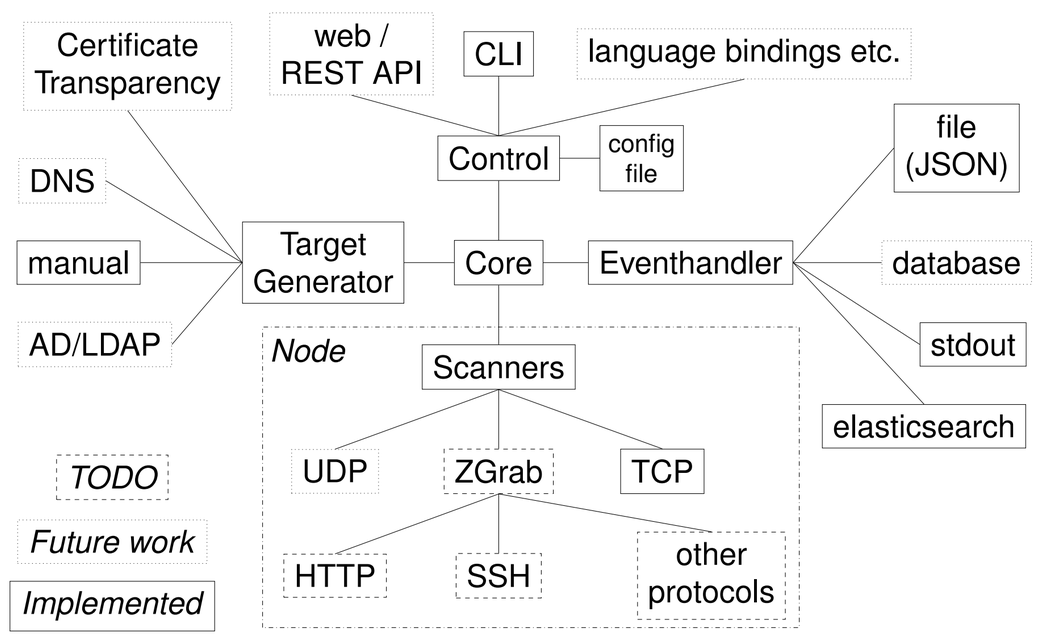
Background

- Nmap highly accurate, but slow when packet loss happens (e.g. firewalls dropping packets instead of rejecting them) → Not sufficient for mid to large size networks
- ZMap/Masscan really fast on high-throughput links (vertical scaling), but inadequate results in case of packet loss → Not sufficient for non-Internet networks (mid to lower size, central bottlenecks)
- Maintained, flexible and stable frameworks for distributed port scanning across arbitrary scanner nodes (horizontal scaling) do not exist

Prototype

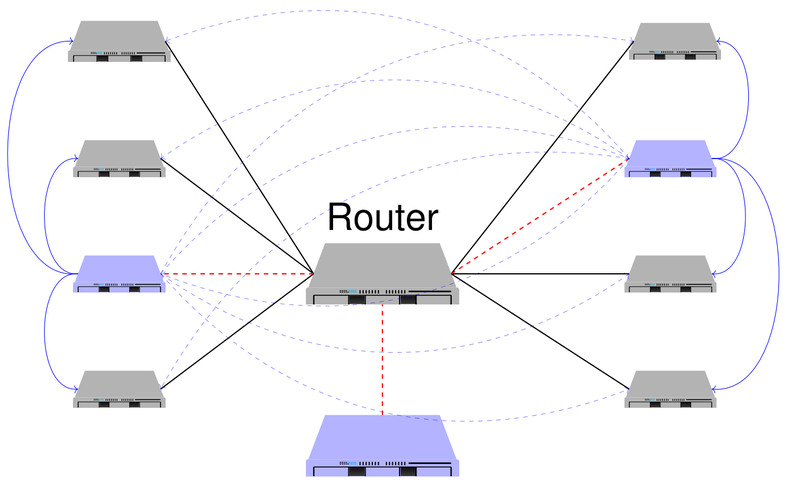
- Implemented in Go, no runtime dependencies
- Server and client binary running without elevated privileges
- Platform independent and embeddable on top of any existing infrastructure
- Scan speedup by distribution of scan targets accross nodes
- Redundant scanning by having targets scanned by same nodes
- Write results to JSON, elasticsearch, stdout
- extensive configuration, highly documented source code
- TLS 1.2 support for communication between node and server
- TLS mutual authentication support
- *Planned/WIP:* ZGrab2 [4] integration
- *Planned:* Rate limiting and UDP support

Architecture



Measurements

1. Reproducible mininet setup



2. Comparison against other scanners
   - Perform scan with Nmap, Zmap and Masscan and our prototype against RBG network
   - Compare results: Scan time, discovered ports, result variations etc.
3. Comparison: Visibility inside and outside of RBG network
   - Place scanning nodes inside and outside of RBG network and perform a scan
   - Compare results and maybe identify firewall misconfigurations

Planned Schedule

Short time schedule

- Official start date: October 15, 2018
- Official end date: April 15, 2019
- Weeks left: 11.5

Progress overview

- Software architecture is defined and works
- Implementation of core functionality finished, features are currently implemented
- Evaluation phase is being planned and prepared

Future work

- Write results to SQL database backend
- SSH-like/Curve authentication (no certificates/PKI required)
- Improved automation, e.g. scan task submission via REST API
- Other scanner backends (e.g. ZMap, Nmap, Masscan) on nodes

Note: I plan to release and continue to work on the prototype after finishing the thesis

[1] Censys. https://censys.io/.
[2] Go supported platforms. https://golang.org/doc/install/source#environment.
[3] Shodan. https://www.shodan.io/.
[4] Zgrab2. https://github.com/zmap/zgrab2.