

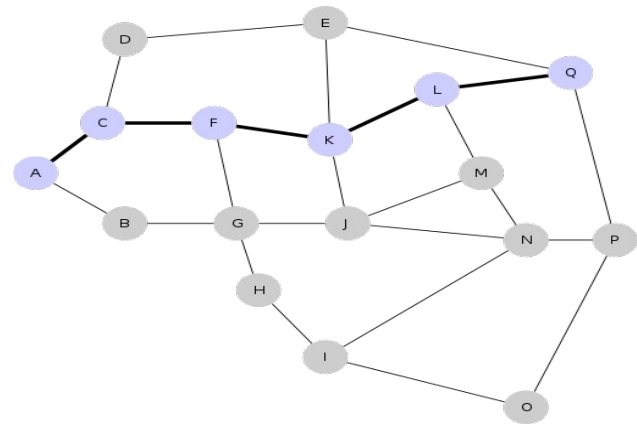


Old and New Ad-Hoc Mesh Protocols



What for?

- Mesh Networks
 - potential loops
- Mobile
 - topology changes
- Ad-hoc
 - no existing infrastructure





Routing Software

- Algorithms & Protocols & Implementation
 - Research
 - Commercial
 - Mesh Routing Communities
 - Focus :-)

DSDV

- Destination-Sequenced Distance Vector routing
 - proactive
- C. Perkins and P. Bhagwat in 1994
- distributed version of Bellman-Ford
- Academic

from the view of node A

Destination	Next Hop	Number of Hops	Sequence Number	Install Time
A	A	0	A 34	00200
B	B	1	B 45	00220
C	B	2	C 28	00260

AODV

- Ad-hoc On-Demand Distance Vector Routing
 - reactive
- mostly academic
- 2003 Nokia
- implementations
 - MAD-HOC, AODV-UU, AODV-UCSB, ...
- used by  ZigBee®



OLSR (1)

- Open Link State Routing
- INRIA 2004
- algorithm
 - Link State
 - MPR (Multi Point Relay)
- Used by e.g. Freifunk Berlin



OLSR

OLSR / olsrd

- active 2004-2019
- last release 0.9.8
- programmed in C

- \$ git shortlog -s -n
- 2377 Ferry Huberts
- 845 Henning Rogge
- 685 Andreas Tonnesen
- 682 Bernd Petrovitsch
- 323 Hannes Gredler
- 280 Sven-Ola Tuecke
- 232 Thomas Lopatic
- 155 Markus Kittenberger
- 80 Hans-Christoph Steiner
- 80 Vasilis Tsiligiannis
- 47 Saverio Proto



OLSR / OONF

- OLSR.org Network Framework (OONF)
- `$ git shortlog -s -n`
 - 1479 Henning Rogge
 - ...
- 2011-2018
- v0.9.2
- programmed in C



batmand (1)

- Thomas Lopatic + Elektra Wagenrad
 - answer to OLSR
 - Hands-on approach
- 2006-2011
- Algorithm
 - Layer 3
 - Distance Vector

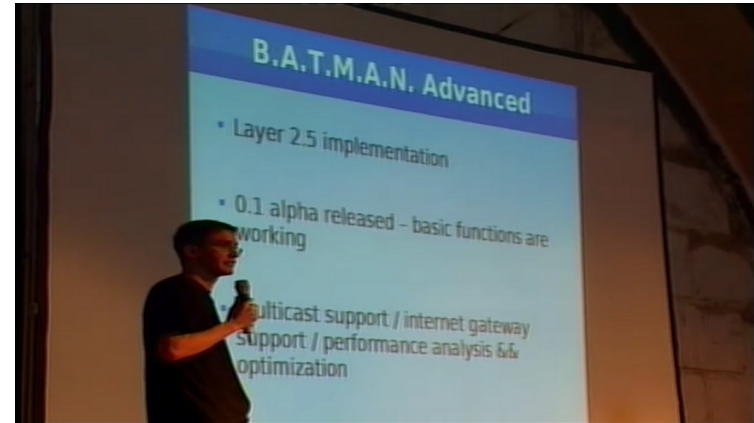
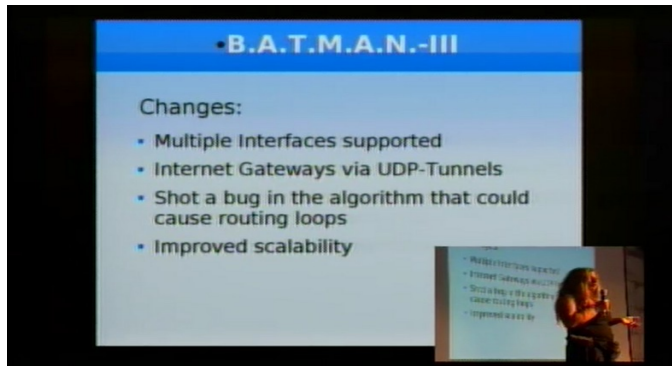


batmand (2)

- <https://git.open-mesh.org/batmand.git>
- `$ git shortlog -s -n`
 - 416 Marek Lindner
 - 86 Sven Eckelmann
 - 85 Andreas Langer
 - 62 Simon Wunderlich
 - 36 Axel Neumann
 - 24 Stefan Sperling
 - 11 Mirko Vogt
 - 11 Elektra Wagenrad

batmand (3)

- Website
 - <https://www.open-mesh.org/projects/batmand/wiki>



B.A.T.M.A.N.-adv (1)

- Better Approach To Mobile Ad-hoc Networking
- Part of the Linux Kernels
 - Kernel module
- Used by >95% of Freifunk communities
- Algorithm
 - Distance Vector
 - bandwidth metric
 - Layer 2 (network bridge)





B.A.T.M.A.N.-adv (2)

- Ab 2007 as Kernel Modul
 - No IP Addresssen => MAC-Adressen
 - IP independent
 - switched to bandwidth metric in BATMAN_V
 - Commercial use
- IPv4, IPv6 + Multicast Groups
- Programmed in C
- Huge Ethernet switch

B.A.T.M.A.N.-adv (3)

- git shortlog -s -n

- 1064 Sven Eckelmann *
- 740 Marek Lindner
- 449 Simon Wunderlich *
- 404 Antonio Quartulli
- 172 Linus Lüssing *
- 46 Andrew Lunn
- 37 Martin Hundebøll
- 24 Matthias Schiffer
- 21 Markus Pargmann

- 15 Andreas Langer
- 12 Linus Luessing
- 11 Joe Perches
- ...

* work in the same a company and maintain batman-adv 😊



B.A.T.M.A.N.-adv (4)

- Webseite:
 - <https://www.open-mesh.org/projects/batman-adv/wiki>

BMX6 (1)

- Axel Neuman
- BatMan eXperimental
 - bmx6 (aka BMX6)
 - 2006 - 2011
 - bmx7
 - 2011-2019
- Algorithm
 - based on distance-vector routing





BMX6 (2)

- Isolate node properties into single node description (e.g. addresses, name, networks)
- Propagate node description once and reference it via its hash (e.g. from routing updates)
- Lots of switches

BMX7 (1)

- RSA-signed node descriptions and routing updates
- Authenticated node IDentities
- Ownership proving IPv6 addresses
 - Crypto generated
- Secure routing against untrusted nodes
- Capacity and interference aware routing metric



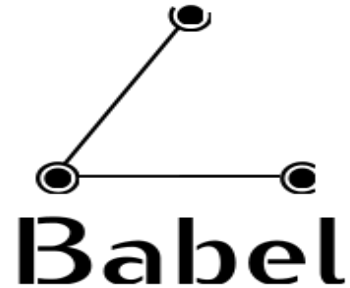


BMX6 / BMX7

- <https://bmx6.net> (defunct)
- <https://github.com/bmx-routing/>

Babel (1)

- Juliusz Chroboczek und others
 - Paris-Diderot (Paris 7)
- Algorithm
 - proactive / Distance Vector
 - inspired by DSDV, AODV, EIGRP
- implementation
 - Layer 3
 - IPv4+IPv6





Babel (2)

- Development
 - Started in 2007
 - Also used for commercial projects
- Code
 - Programmed in C
 - User space daemon



Babel (3)

- Website
 - <http://www.irif.fr/~jch/software/babel/>
- “Babel Doesn’t Care”
 - BattleMeshV8 (2015)
 - <https://www.youtube.com/watch?v=1zMDLVIn3XM>
- “Evolution of the Babel Routing Protocol”
 - BattleMeshV12 (2019)
 - <https://www.youtube.com/watch?v=Mflw4BukshQ>

Yggdrasil (1)

- People
 - Neil Alexander & Arceliar
- Experiment in/from Matrix/Element Community
 - Spanning Tree + DHT
 - Source Routing
 - Hop Count Metric
- Inspired by CJDNS:
 - suffered by performance and scalability



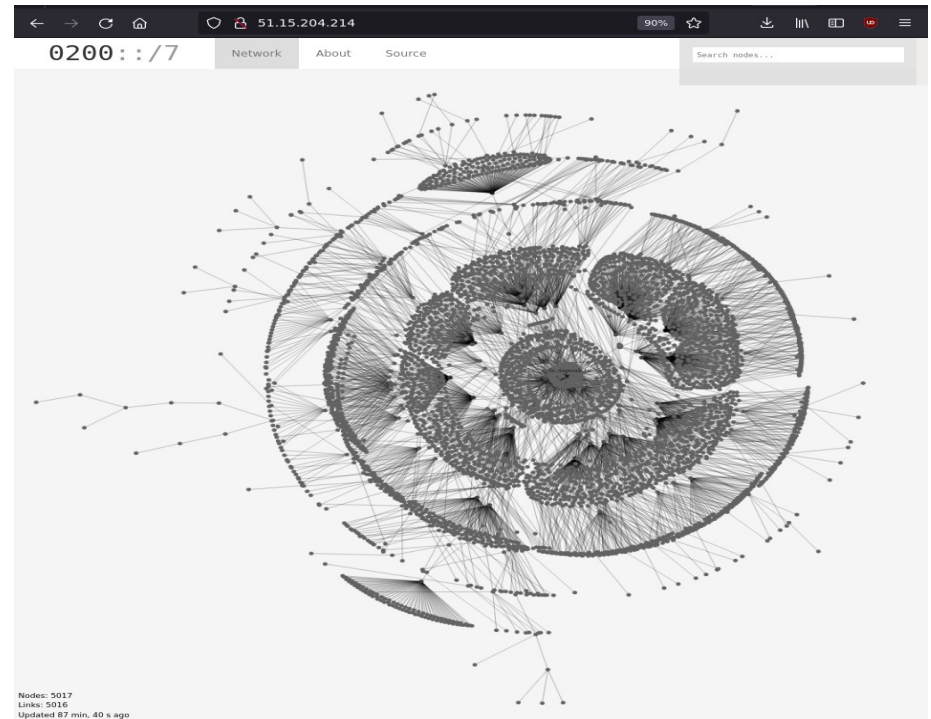


Yggdrasil (2)

- Code
 - implemented in Go
 - Layer 3 / IPv6 only
 - uses big MTUs
- Development started begin of 2018
- Address space 0200::/7 (“unused”)
- End-to-Ende Crypto
 - IPv6 Address from public key

Yggdrasil (3)

- one spanning tree
- network of ~5000 nodes
 - As of 01.05.2022
 - <http://51.15.204.214/>
- overlay over the Internet
 - public peer lists
- can be used by WiFi 802.11s
 - without mesh (mesh_fwding=0)
- local multicast auto peering



Yggdrasil (4)

0.3.x

- spanning tree
- DHT as a circle (Chord)
- curve25519 keys
- root has highest key

0.5.x (in development)

- replace source routing with greedy tree routing

0.4.x

- SNEK DHT
- ed25519 keys
- root has lowest key
- “[...] tree routing is largely only used for bootstrapping DHT paths and determining source routes.”
- “Session traffic will now use source routing if available, to ensure that the overall connection quality of sessions is preserved. If a source-routed path fails, the traffic will revert to DHT forwarding seamlessly.”



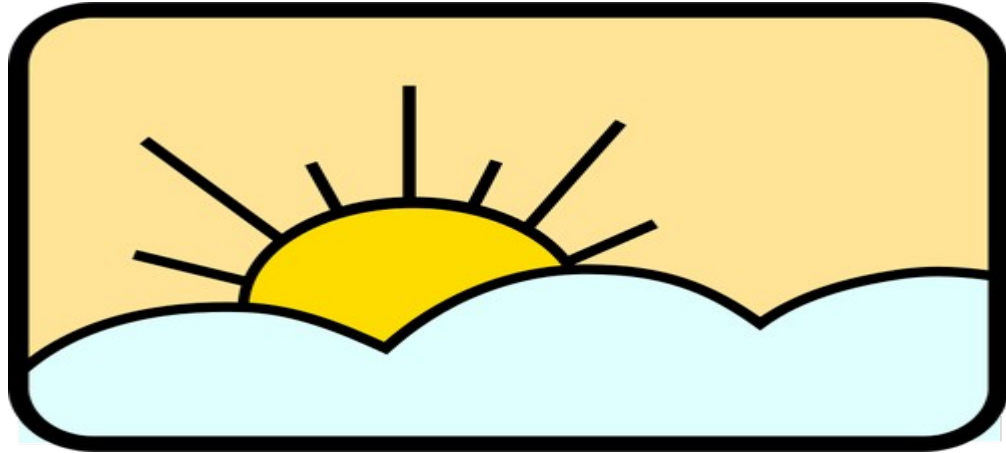
Yggdrasil (5)

- Website: <https://yggdrasil-network.github.io/>
- Recommended:
 - “Growing Pinecones for P2P Matrix” (2022)
 - FOSEM 2022
 - https://fosdem.org/2022/schedule/event/matrix_p2p_pinecone/
 - “Pinecones and Dendrites - P2P Matrix Progress”
 - FOSDEM 2021
 - https://archive.fosdem.org/2021/schedule/event/matrix_pinecones/



End / Start

- Thank you



- Discussion