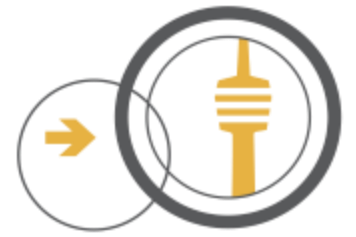


ffs-Monitor

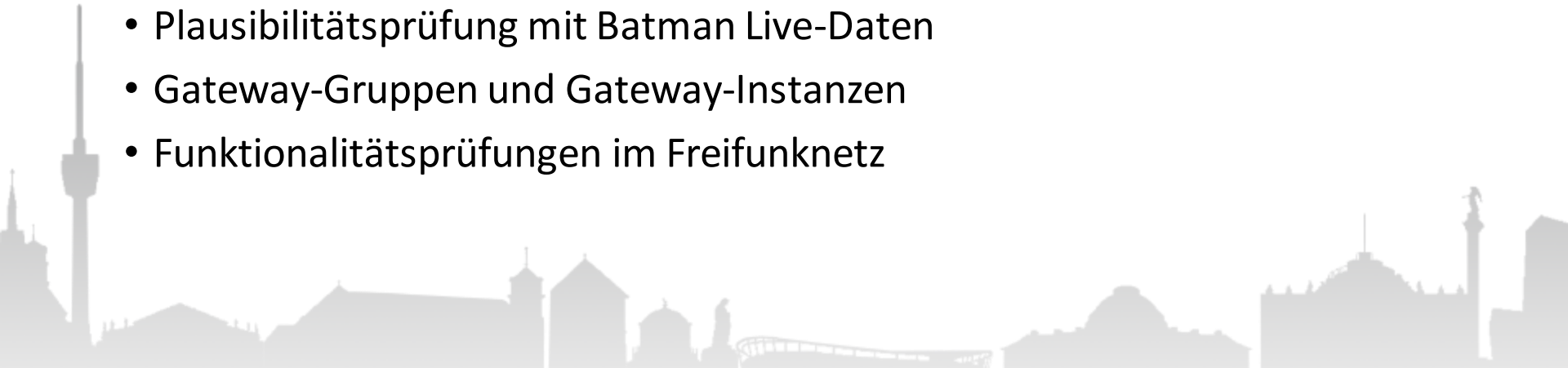
Qualitätsmanagement
im Freifunknetz



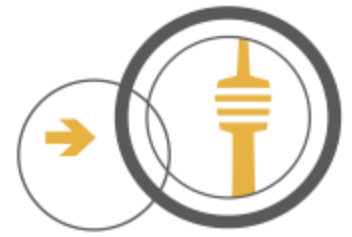
ffs-Monitor - QM im Freifunknetz



- Entstehungsgeschichte – Anlass für QM
- Statusdaten der Nodes im Freifunknetz
- Zuverlässigkeit der Daten – Redundanz
- Abhängigkeiten zwischen den Daten
- VPN-Uplink zwischen Node und Gateway
- VPN-Statusdaten der Gateways
- Plausibilitätsprüfung mit Batman Live-Daten
- Gateway-Gruppen und Gateway-Instanzen
- Funktionalitätsprüfungen im Freifunknetz

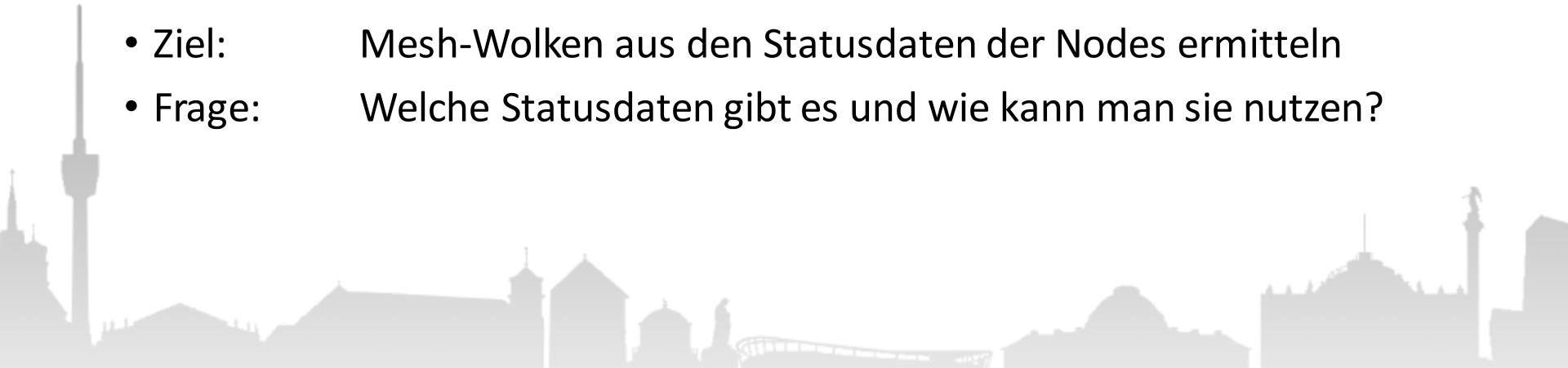


ffs-Monitor - QM im Freifunknetz



Entstehungsgeschichte

- Anlass: Aufteilung des Freifunk-Netzes in Teilnetze (Segmentierung)
- Aufgabe: Zuordnung der Nodes zu den Segmenten
- Idee: Segmente an geografischer Lage (Landkreise) ausrichten
- Wichtig: Meshende Nodes müssen im selben Segment sein
- Problem: Nicht alle Nodes haben Koordinaten oder passende Namen
- Ziel: Mesh-Wolken aus den Statusdaten der Nodes ermitteln
- Frage: Welche Statusdaten gibt es und wie kann man sie nutzen?



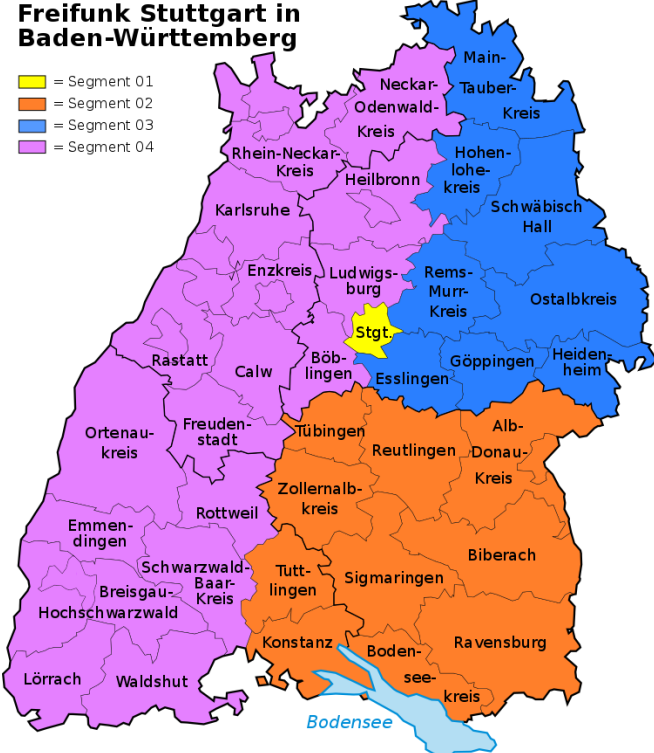
ffs-Monitor - QM im Freifunknetz



Segmente - Stand 05/2016

Freifunk Stuttgart in Baden-Württemberg

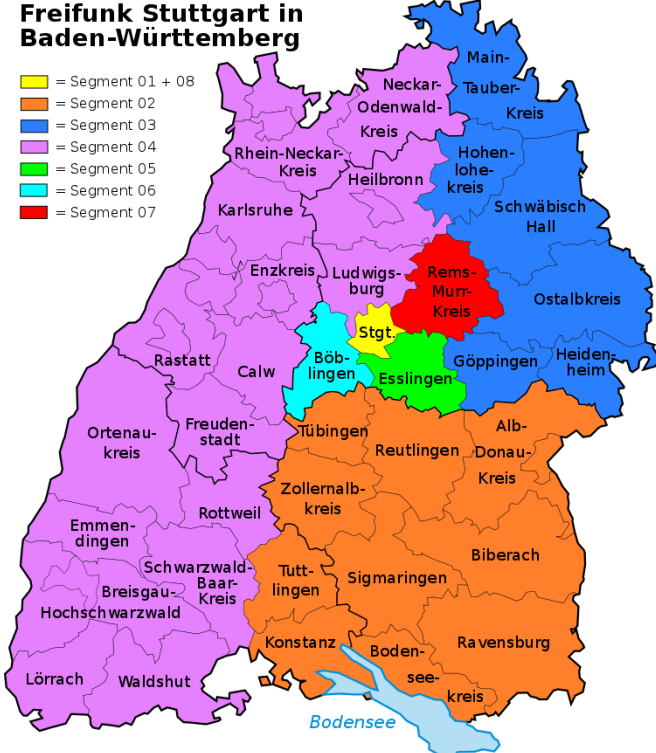
- = Segment 01
- = Segment 02
- = Segment 03
- = Segment 04



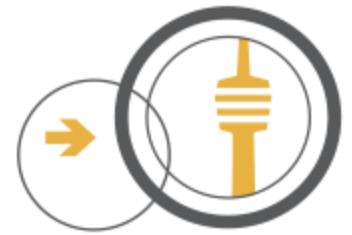
Segmente - Stand 04/2017

Freifunk Stuttgart in Baden-Württemberg

- = Segment 01 + 08
- = Segment 02
- = Segment 03
- = Segment 04
- = Segment 05
- = Segment 06
- = Segment 07



ffs-Monitor - QM im Freifunknetz



Statusdaten von Freifunk-Nodes

Nodeinfo

- Node-ID
- Hostname
- Hardware
- Location
- Network
- Software

Statistics

- Node-ID
- Clients
- Gateway
- Mesh-VPN
- Load (CPU / RAM)
- Traffic
- Uptime

Neighbours

- Node-ID
- Mesh-Interfaces mit
MACs der Nachbar-Nodes





ffs-Monitor - QM im Freifunknetz

Nodeinfo = Alfred 158

30:b5:c2:ee:8a:7a	
hardware	TP-Link TL-WR841N/ND v9
model	1
nproc	
hostname	ffs-Leo-IPRO-GmbH-3
location	
latitude	48,79868389
longitude	9,00381088
network	
addresses	
fd21:b4dc:4b06:0:32b5:c2ff:fee:8a7a	
fd21:711::32b5:c2ff:fee:8a7a	
fe80::32b5:c2ff:fee:8a7a	
mac	30:b5:c2:ee:8a:7a
mesh	
bat0	
interfaces	
other	
4a:d2:37:1f:19:0b	
4a:d2:37:1f:19:0f	
4a:d2:37:1f:19:0a	
node_id	30b5c2ee8a7a
software	
autoupdater	
branch	nightly
enabled	false
batman-adv	
compat	15
version	2016.2
fastd	
enabled	true
version	v18
firmware	
base	gluon-v2016.2.5
release	1.0+2017-04-10-g.84dd03a-s.5cab37c
status-page	
api	1
system	
site_code	ffs

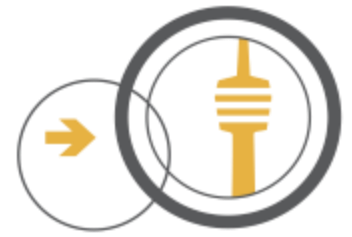
Statistics = Alfred 159

30:b5:c2:ee:8a:7a	
clients	
gateway	02:00:38:06:04:01
gateway_nexthop	02:00:38:06:04:01
idletime	230296,05
loadavg	0,08
memory	
buffers	2188
cached	4808
free	3208
total	28244
mesh_vpn	
groups	
backbone	
peers	
gw01	null
gw02	null
gw03	null
gw04	92073,322
gw05	null
gw06	null
gw07	null
gw08	null
gw09	null
gw10	null
node_id	30b5c2ee8a7a
processes	
running	5
total	53
rootfs_usage	0,6458
traffic	
forward	
bytes	2841273327
packets	4012654
mgmt_rx	
bytes	464573670
packets	3895671
mgmt_tx	
bytes	684768196
packets	4976864
rx	
bytes	2313491504

Neighbours = Alfred 160

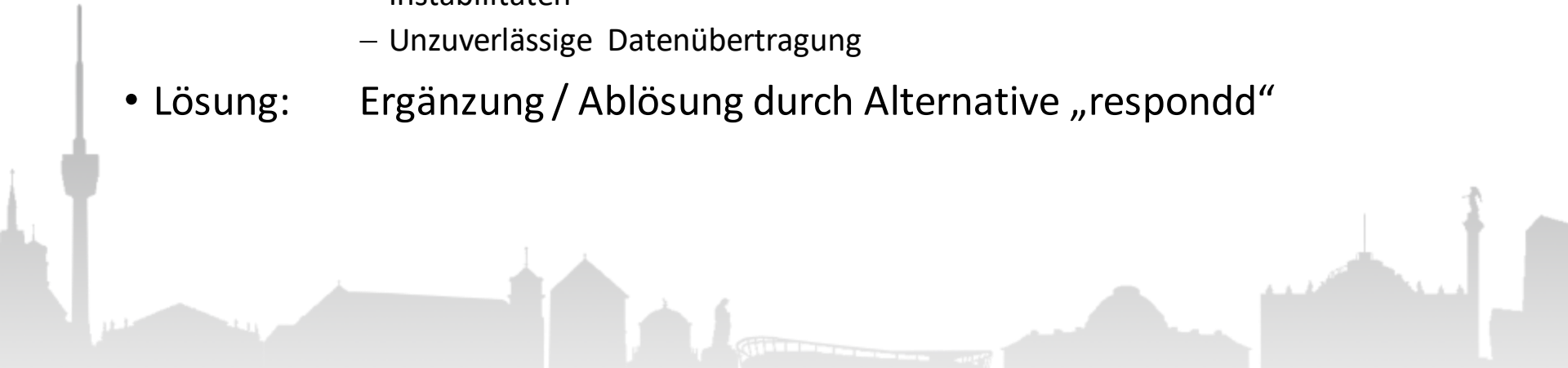
30:b5:c2:ee:8a:7a	
batadv	
4a:d2:37:1f:19:0a	
neighbours	
ae:bb:0c:e7:ab:2a	
lastseen	4,56
tq	249
4a:d2:37:1f:19:0f	
neighbours	
02:00:38:06:04:01	30b5c2ee8a7a
node_id	30b5c2ee8a7a
wifi	
4a:d2:37:1f:19:0a	
neighbours	
ae:bb:0c:e7:ab:2a	
inactive	10
noise	-95
signal	-70

ffs-Monitor - QM im Freifunknetz

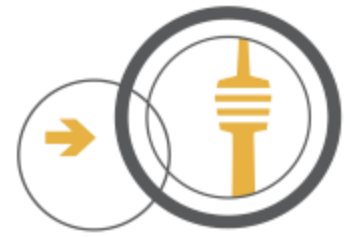


A.L.F.R.E.D.

- Idee: Dezentrale Datenerfassung
- Konzept: Daten werden jeweils zum nächstgelegenen Server gesendet, die Server synchronisieren sich untereinander.
- Problem: Programmfehler, die nicht korrigiert werden
 - Nur ein Alfred-Server möglich
 - Instabilitäten
 - Unzuverlässige Datenübertragung
- Lösung: Ergänzung / Ablösung durch Alternative „respondd“



ffs-Monitor - QM im Freifunknetz



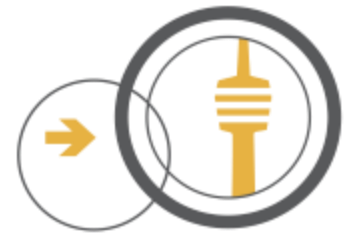
respondd

- Abruf per UDP Multicast-Message
- Mehrere unabhängige Server möglich
- HopGlass-Server zur Datenaufbereitung
- Inzwischen Basis für den MeshViewer
- Daten als „raw.json“ für ffs-Monitor

raw.json

Field	Value
firstseen	2017-02-24T12:43:25.359Z
lastseen	2017-04-22T13:14:45.436Z
lastupdate	
statistics	2017-04-22T13:14:45.436Z
nodeinfo	2017-04-22T13:00:42.245Z
neighbours	2017-04-22T13:12:47.382Z
statistics	
clients	
traffic	
gateway	02:00:38:06:01:03
gateway_nexthop	02:00:38:06:01:03
mesh_vpn	
node_id	30b5c2ee8a7a
rootfs_usage	0,6458
memory	
uptime	916736,76
idletime	813472,23
loadavg	0,05
processes	
nodeinfo	
software	
network	
addresses	
mesh	
mac	30:b5:c2:ee:8a:7a
location	
owner	
system	
node_id	30b5c2ee8a7a
hostname	ffs-Leo-IPRO-GmbH-3
hardware	
neighbours	
batadv	
wifi	
node_id	30b5c2ee8a7a

ffs-Monitor - QM im Freifunknetz



MAC-Adressen der Interfaces

- Node-ID entspricht der Primären MAC ohne „:“ = Client-MAC
- MACs der Node-Interfaces werden nach festen Regeln festgelegt
 - bis Gluon 2016.1.x vordere 3 Bytes modifiziert, hintere 3 Bytes identisch
 - ab Gluon 2016.2 MD5-Hash, hintere 3 Bit kennzeichnen Interface

```
bat0 Link encap:Ethernet HWaddr 30:B5:C2:EE:8A:7A
inet6 addr: fe80::32b5:c2ff:feee:8a7a/64 Scope:Link
```

```
br-client Link encap:Ethernet HWaddr 30:B5:C2:EE:8A:7A
inet6 addr: fd21:b4dc:4b06:0:32b5:c2ff:feee:8a7a/64 Scope:Global
inet6 addr: fd21:711::32b5:c2ff:feee:8a7a/64 Scope:Global
inet6 addr: fe80::32b5:c2ff:feee:8a7a/64 Scope:Link
```

```
br-wan Link encap:Ethernet HWaddr 4A:D2:37:1F:19:08
inet addr:10.200.1.240 Bcast:10.200.1.255 Mask:255.255.255.0
inet6 addr: fe80::48d2:37ff:fe1f:1908/64 Scope:Link
```

```
client0 Link encap:Ethernet HWaddr 4A:D2:37:1F:19:08
inet6 addr: fe80::48d2:37ff:fe1f:1908/64 Scope:Link
```

```
eth0 Link encap:Ethernet HWaddr 30:B5:C2:EE:8A:7A
```

```
eth1 Link encap:Ethernet HWaddr 30:B5:C2:EE:8A:7B
```

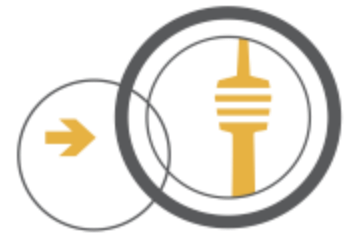
```
ibss0 Link encap:Ethernet HWaddr 4A:D2:37:1F:19:0A
inet6 addr: fe80::48d2:37ff:fe1f:190a/64 Scope:Link
```

```
local-node Link encap:Ethernet HWaddr 02:00:0A:25:00:01
inet addr:172.21.24.254 Bcast:172.21.63.255 Mask:255.255.192.0
inet6 addr: fe80::aff:fe25:1/64 Scope:Link
inet6 addr: fd21:711::1/128 Scope:Global
```

```
mesh-vpn Link encap:Ethernet HWaddr 4A:D2:37:1F:19:0F
inet6 addr: fe80::48d2:37ff:fe1f:190f/64 Scope:Link
```

```
primary0 Link encap:Ethernet HWaddr 4A:D2:37:1F:19:0B
```

ffs-Monitor - QM im Freifunknetz



Abhängigkeiten zwischen den Daten

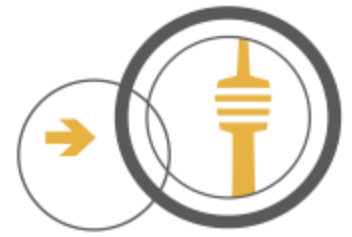
- Node selbst kennt seine Segmentzuordnung nicht
- Segmentzuordnung erfolgt über den VPN-Key auf den Gateways
- VPN-Key wird nicht mit den Datensätzen der Nodes übertragen
- VPN-Keys werden bei der Registrierung unabhängig im Git abgelegt

#MAC: 30:b5:c2:ee:8a:7a

#Hostname: ffs-Leo-IPRO-GmbH-3

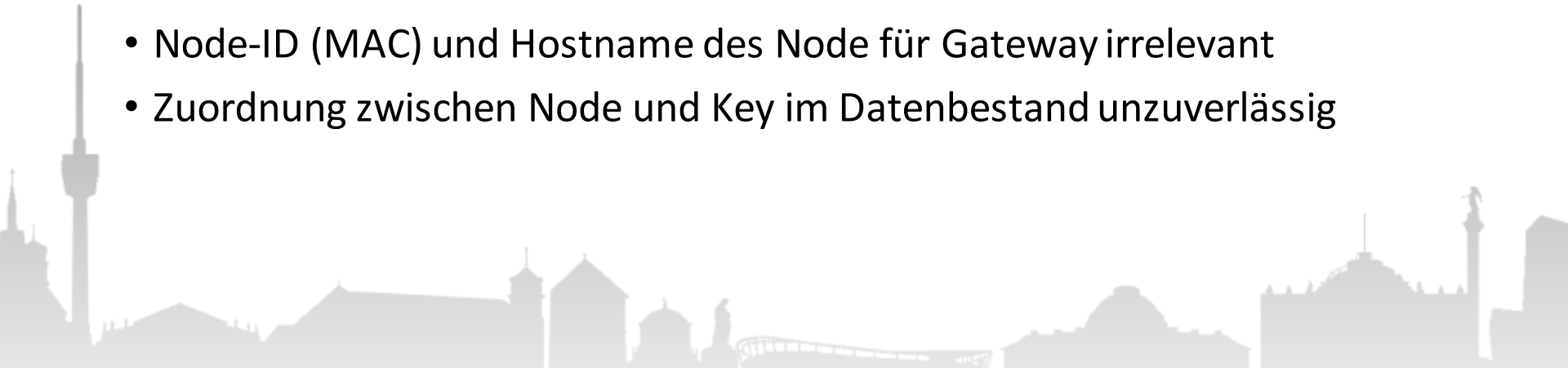
key "11aabc8fdf701cf1c4b1f2fb7ab1468ea464dafb7eda5b22ef6a0975a86e89f5";

ffs-Monitor - QM im Freifunknetz

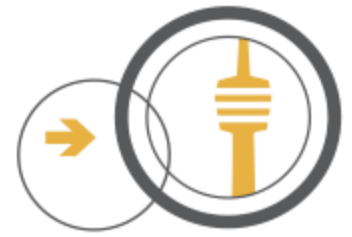


VPN-Uplink zwischen Node und Gateway

- Node enthält Liste aller Gateways und möglicher Segmente
- VPN-Verbindung zwischen Nodes und Gateways mittels „fastd“
- Segment-Zugang auf dem Gateway über unterschiedliche Ports
- Node sendet seinen public Key mit dem Verbindungswunsch
- Gateway prüft den Key auf Gültigkeit im Segment
- Node-ID (MAC) und Hostname des Node für Gateway irrelevant
- Zuordnung zwischen Node und Key im Datenbestand unzuverlässig



ffs-Monitor - QM im Freifunknetz

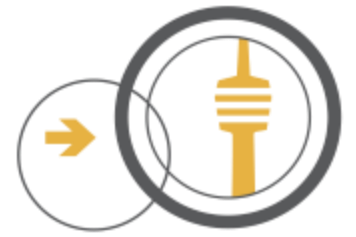


VPN-Statusdaten der Gateways

- Bei bestehender VPN-Verbindung ist Absende-MAC der Daten im Tunnel Bestandteil des fastd-Status auf dem Gateway
- Gluon-Node sendet batman-adv-Daten (Mesh) durch den Tunnel
- Gateway sieht MAC vom mesh-vpn Interface des Node
- VPN-Statusdaten werden als json-Dateien zur Verfügung gestellt

11aabc8fdf701cf1c4b1f2fb7ab1468ea464dafb7eda5b22ef6a0975a86e89f5	
name	ffs-30b5c2ee8a7a
connection	
established	425
method	salsa2012+umac
statistics	
mac_addresses	
4a:d2:37:1f:19:0f	

ffs-Monitor - QM im Freifunknetz

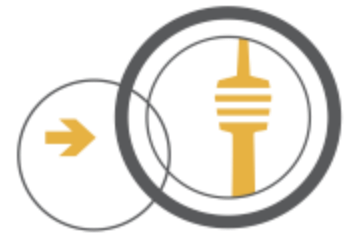


Plausibilitätsprüfung mit Batman Live-Daten

- Tabellen mit den Verwaltungsdaten von batman-adv
 - Gateway-Liste
 - Originator-Table
 - Globale Translation-Table
 - Lokale Translation-Table
- Traceroute zur Verbindungsverfolgung



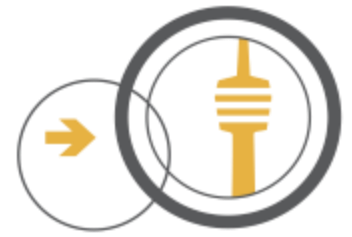
ffs-Monitor - QM im Freifunknetz



Gateway-Gruppen und Gateway-Instanzen

- Nodes haben nur 10 Gateway-Einträge pro Segment
- Ziel: Mehr Nodes ermöglichen und bessere Lastverteilung
- Lösung: Eintrag im Node entspricht Gateway-Gruppe
 - gw01s05 = Gateway-Gruppe 01 im Segment 05
 - gw04s03 = Gateway-Gruppe 04 im Segment 03
- Ein bis 10 tatsächliche Gateways = Instanzen pro Gateway-Gruppe
 - gw01n03 = Gateway-Instanz 03 in Gruppe 01
 - gw04n01 = Gateway-Instanz 01 in Gruppe 04
- Jede Gateway-Instanz kann ein oder mehrere Segmente bedienen
- Dynamisches Rein- und Rauschalten von Instanzen in die Gruppen via DNS

ffs-Monitor - QM im Freifunknetz

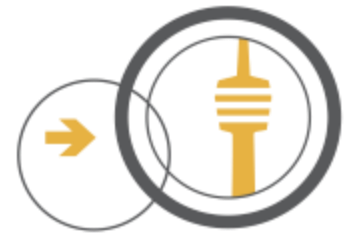


Funktionalitätsprüfungen im Freifunknetz

- Jedes Segment hat mindestens zwei GW-Instanzen aus verschiedenen Gruppen
- Keine Segment-Kopplungen („Kurzschlüsse“)
- Nodes befinden sich im richtigen Segment
- DNS funktioniert auf den Gateways
- DHCP funktioniert
- Updateserver ist erreichbar
- Mindestens ein NTP-Server liefert korrekte Zeit



ffs-Monitor - QM im Freifunknetz



Auszüge aus dem Protokoll des ffs-Monitor

Checking DNS for Gateways: gw.freifunk-stuttgart.de ...

```
Seg.00 -> [gw05n01', 'gw07']
Seg.01 -> [gw05n02', 'gw08n00']
Seg.02 -> [gw05n02', 'gw08n00']
Seg.03 -> [gw05n03', 'gw08n00', 'gw08n04', 'gw08n05']
Seg.04 -> [gw05n03', 'gw08n00']
Seg.05 -> [gw01n03', 'gw08n00']
Seg.06 -> [gw01n03', 'gw08n00']
Seg.07 -> [gw01n03', 'gw08n00', 'gw08n03']
Seg.08 -> [gw01n03', 'gw08n00']
Seg.09 -> [gw01n03', 'gw04n01', 'gw08n00']
Seg.10 -> [gw01n03', 'gw04n01', 'gw08n00']
```

Analysing alfred-json-160.json ...

```
++ Neighbour MAC unknown: 1 V 68:2a:a8:69:12:30 = b'ffs802aa8691230' -> 40:4d:7f:6b:9f:88
++ Neighbour MAC unknown: 3 V 68:72:51:4c:4d:2a = b'ffs-geislingen-camp-2' -> a0:75:91:c4:58:d7
++ Neighbour MAC unknown: 1 64:70:02:de:b9:78 = b'ffs-Forst71-10G' -> d8:96:95:5b:00:3e
++ Neighbour MAC unknown: 1 V 10:fe:ed:e6:20:9c = b'ffs-Forst71-Uplink' -> d8:96:95:5b:00:3e
```

```
>>> Segment set by ZIP-Code: 00:90:dc:06:38:62 = b'ffs-0090dc063862' -> 70565 -> 8
!! Invalid Location: ec:08:6b:61:6e:d6 = b'ffs-AL-BU-03' -> 48.34860063 | 48.34860063
>>> Segment set by ZIP-Code: ec:08:6b:72:f9:2e = b'Kopp' -> 73207 -> 5
>>> Segment set by ZIP-Code: 00:19:99:9a:e7:e5 = b'71522-willy-brandt-platz-2-00-uplink' -> 71522 -> 7
>>> Segment set by ZIP-Code: 68:72:51:82:a9:cf = b'ffs-ditzingen-hirsch' -> 71254 -> 4
!! Invalid Location: ec:08:6b:3f:cc:16 = b'ffs-goepingen-papettino' -> 48.70411615 | 48.70411615
>>> Segment set by ZIP-Code: ec:08:6b:3f:cc:16 = b'ffs-goepingen-papettino' -> 73033 -> 3
```