nodewatcher
deployment and monitoring system

Mitar Milutinović, Jernej Kos

wlan ljubljana – September 27, 2010
Summary.

nodewatcher: planning, deployment and monitoring system for community/diverse/open/chaotic city/country-wide network.
wlan ljubljana – open wireless network in Ljubljana, Slovenia
http://wlan-lj.net

wlan slovenija – open wireless network in Slovenia
http://wlan-si.net
Network.

Mesh network.

Wireless mesh network.
We use every way of connecting nodes.

WiFi, ethernet, tunnels, fiber...
Adaptive, dynamic.

Routing protocol.

OLSR.
Community based.

Everybody can participate.
A common network.

Content.
Services.
Users.

Wireless. Mobile.
Open source.

Linux.

OpenWrt.

http://www.openwrt.org/
In the beginning... 

...there was chaos!
Everything by hand.

Node status.
IP allocation.
Map.
Node configuration.
Basic problems:

Small number of active people.
Small amount of free time
(which we are using for the network).
Especially:

Mostly routine things.
Duplication of work.
Uncoordinated. Confusion.
Hardly approachable for new participants.

Limited only to technically skilled.

And persistent.
So at the start of 2009 we decided, that we need a different approach.
For new users

- Easy approach
- Technical skill not required
- Fast startup
- Not time-consuming
- Uniform
- Clear
- Informative
- Look into the network
- Web site
For existing users

Removal of routine work

Prevent work duplication

Transfer good practices

Self-documentation

Network visualization

Errors, mistakes discovery

Easy maintenance
Along came a nodewatcher.

Automate as much as possible.
If technology knows how to do something, it should do it.
live version: https://nodes.wlan-lj.net
development: http://wlan-lj.net/wiki/Podrobnosti/Nodewatcher

Jernej Kos
Luka Čehovin
Mitar Milutinović

And others.
web interface

basic models (Django)

image generator

beanstalkd queue

gwpolicyd

tc

monitor

routing (OLSR)
Planning.

Deployment.

Monitoring and maintenance.
Planning.

IP allocations.
Interfaces: subnets, DHCP, VPN.
Position on the map.
And other node properties.
Deployment.

Image generator.

Or just reference configuration.
from beanstalkd queue

buildroot compiled for different architectures/platforms

generator configuration → OpenWrt buildroot

bin, trx, lzma → flashable image (firmware)
Deployment.

Plug-and-mesh.

There is no web-based configuration on nodes.
Monitoring and maintenance.

OLSR status.
Internal node status.
monitor

worker 1

worker 2

worker n

routing (OLSR)

WiFi mesh (nodes)

txtinfo

graphs, topology, status, events,...
Monitoring and maintenance.

HTTP, simple format.

No SNMP.
general.version: 2.0b.r796
general.local_time: 1259248995
general.uptime: 1334081.18 1283631.13
general.loadavg: 0.02 0.03 0.00 1/31 7409
general.memfree: 2512
general.buffers: 0
general.cached: 2728
wifi.bssid: 02:CA:FF:EE:BA:BE
Extendable.

Robust.

Accessible.
Aggregation.

Analysis.

Presentation.
Monitoring and maintenance.

WAN/VPN policy.
Monitoring and maintenance.

Events.

E-mail notifications, RSS.
nodewatcher.

Not so easy to implement.

Right.
WAN/VPN policy for every node.

Parallelization.

Robustness.
Future.

User experience.

Not more technical than really necessary.
But open and hackable.

A platform/framework.

You can still add web-based configuration to your node. If you want.
Example.

Solar node.
How things look like?
Live presentation.
Future.

Mesh long and prosper!
Questions?

mitar@tnode.com, kostko@unimatrix-one.org

http://wlan-lj.net