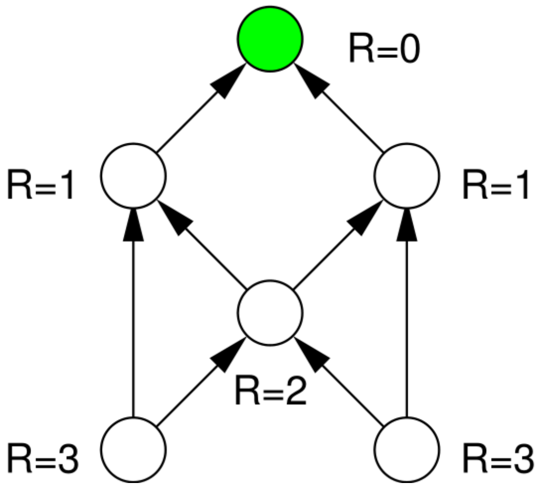
Keeping the DODAG consistent

Ranks

- As we have seen, nodes must choose a rank that is greater than the rank of its parent(s)
- This is also true if a node needs (or wants) to find a new parent, for example when:
  - New nodes appear
  - Old nodes disappear
  - Signal strength changes



- However, a node is only allowed to choose a parent with a lower rank than its own current rank
  - Should avoid that a node becomes the child of one of its children (→ loop)

Repairing the DODAG

- Loop avoidance is a big issue, especially in lossy wireless networks
- Despite ranks, loops can still happen if DIO or DAO messages are lost
- If a problem has been detected, the DODAG must be repaired. Two ways:
  - Local repair: done by individual nodes
  - Global repair: done by the root node

- There are several mechanisms in RPL to detect problems. Details differ between implementation.
- Same for repairs: different implementations possible. Some implementations might not support global repair.

Detecting routing problem

- RFC 6553 defines IPv6 header options for RPL that contain additional information to detect routing problems
- Direction flag: indicates the expected direction (up or down) of a data packet
  1. Sender sets the direction flag
  2. If an up-packets is forwarded from a node A to a node B with higher rank, a problem is detected by B (same for down-packets forwarded to lower ranks)
  3. Node B initiates a local repair

Local repair

- A node might trigger a local repair if it detects a routing problem, e.g.
  - has lost its parent
  - received a packet with a wrong direction flag
  - …
- To do a local repair, the node
  1. detaches itself (and its children) from the DODAG by advertising a rank of INFINITE_RANK (255) (“route poisoning”)
  2. sends DIS messages to find a new parent
  3. select a new parent

Global repair

- Version numbers: When the root advertises a DODAG with the DIO message, it also includes a version number
- The root can decide to increase the version number and advertise a new version of the DODAG (“global repair”)
  - During the repair process, two version of a DODAG temporarily exist
  - A node that joins a DODAG with version *X* ignores lower versions
  - When a node receives a message with higher version, it can move to that new version
    -> choose new parent and new rank

Trickle Timer in RPL

DIO messages

- DIO messages are sent when
  - a new DODAG is constructed (already discussed)
  - a node requests them by a DIS message (already discussed)
  - periodically
- "Unnecessary" DIO messages are avoided by a Trickle timer (RFC 6206)

Trickle Timer

- Trickle algorithm running on each node:
  - T_{min}: minimum duration of the timer
  - T_{max}: maximum duration of the timer
  - T: current duration used by the timer
  - c: number of good messages received by the node
  - k: some threshold for c

Trickle Timer (2)

1. Start with $T \coloneqq T_{min}$ and c:=0
2. Timer is set to a duration t randomly picked from [T/2,T]
3. When t expires and c<k: Send DIO message
4. When T expires:
   - $T \coloneqq 2 \cdot T$ (up to T_{max})
   - Go back to step 2

T_{min}: minimum duration of the timer

T_{max}: maximum duration of the timer

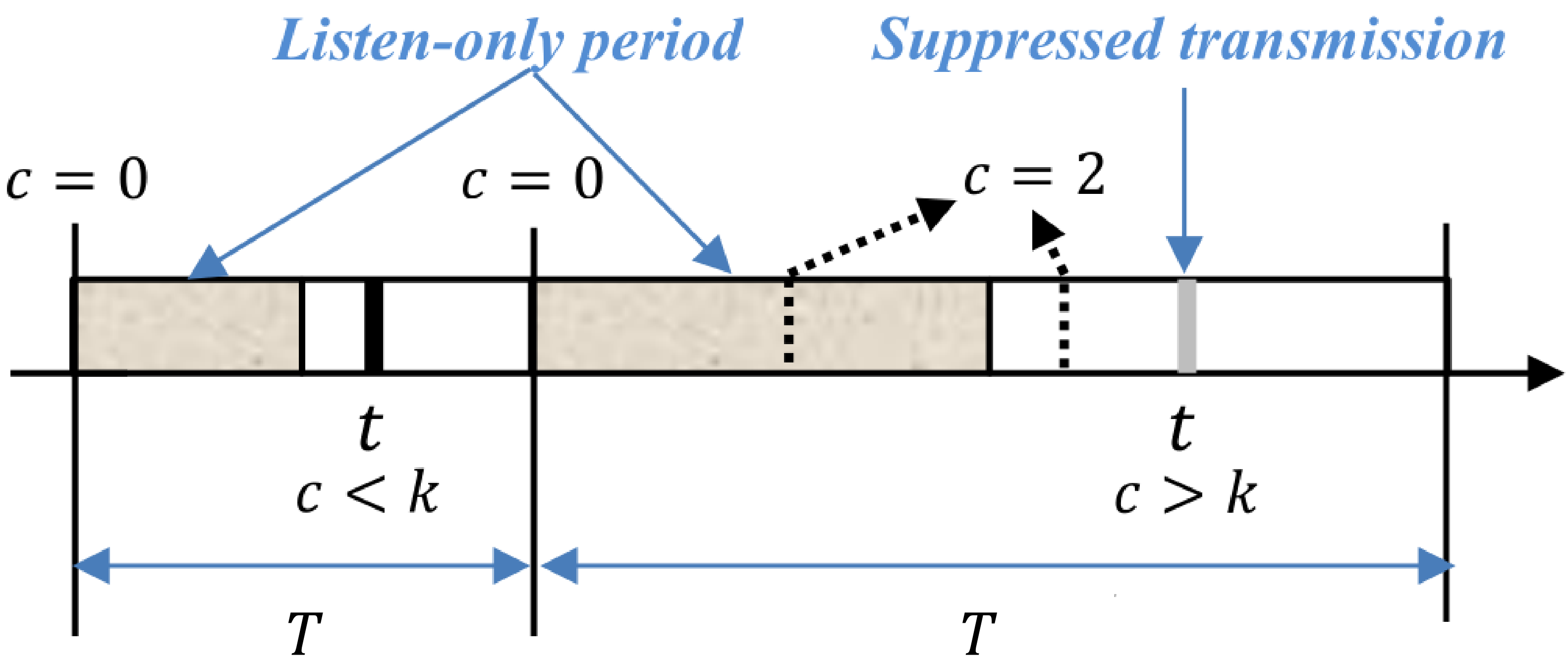
T: current duration used by the timer

c: number of good messages received by the node

k: some threshold for c

- c is increased by 1 for every "good" message
  - Good messages: DIO that does not announce a change in the parents, ranks, etc.
- c is reset to 0 and T is reset to T_{min} for "bad" messages:
  - Bad messages: DIO that announces a change, a data packet with wrong direction flag, new DODAG version, etc.

Trickle Timer Example with k=1



→ Good messages indicate a stable network without problems or changes
→ T is increased
→ Nodes send DIO messages less frequently

Routing Attacks

Selective-Forwarding Attacks

- Attack executed by “malicious” nodes
  - Nodes hacked by the attacker
  - Nodes of the attacker that join the network
- Selective Forwarding attack = Malicious node does not forward all packets
- Most simple form: Malicious node A does not forward packet to victim node V

  → Victim node cannot communicate anymore
- Doesn’t sound too dangerous, right?
  - Easy to detect
  - Nodes can choose a different path if multiple routes available
- Some interesting variations …

Selective-Forwarding: Examples

- Do not forward ACK messages of applications
  Consequence: Retransmissions. Very bad in resource constraint networks like 802.15.4!
- Delay packet forwarding or forward on the wrong path
  Consequence: Creates confused routing information
- Only forward RPL messages and drop all other messages
  Consequence: RPL thinks everything is fine

Selective-Forwarding: Defense

- Difficult to defend against a malicious node inside the network
- Defense
  - Use encryption, so that attacker cannot see what is inside the message (no selective forwarding possible)
  - Measure network quality on application level: if application traffic is lost → notify RPL

Sinkhole Attacks

- A malicious node advertises an artificial very good routing path
  → attracts traffic from nearby nodes, creating a “sinkhole”

- Implementation in RPL: announce very low rank, so that other nodes selects you as parent
  - Will disturb the network, but the DODAG corrects itself after a while (because nodes will choose other parents if link quality is too bad)
  - Defense: use encryption inside PAN

HELLO Flooding

- Attacker node sends a message (“HELLO”) with strong signal power
  → Other nodes will think it is a neighbor node
  → But when sending traffic, the attacker node is out of range

- In RPL, DIO messages can be used for that
  - Again, the network is disturbed but will correct itself after a while
  - Defense: use encryption inside PAN

Wormhole attack

- Attacker creates a fast network connection between two far points of the 802.15.4 network, for example by using WiFi

  → Will become a preferred path for many nodes
- Attacker can
  - Study traffic (eavesdropping)
  - Modify packets
  - Selectively drop packets
  - …

DODAG Version Attack

- Malicious node sends DIO messages with fake new version
  → Nodes will join it and even advertise it in their DIO messages!

- Loops possible because the fake DODAG did not start from the root node

Conclusion

- Routing attacks try to disturb the network by manipulating routing of traffic
- Many attacks can be prevented by using encryption
  - Some not, for example packet-dropping in the Wormhole attack!