Impact of Misbehaviour on QoS in Wireless Mesh Networks

Szymon Szott[1], Marek Natkaniec[1], and Albert Banchs[2]

[1] AGH University of Science and Technology, Krakow, Poland
{szott, natkanie}@kt.agh.edu.pl
[2] Universidad Carlos III de Madrid, Madrid, Spain
banchs@it.uc3m.es

Abstract. This paper analyzes the impact of misbehaviour on QoS provisioning in wireless mesh networks. Misbehaviour occurs when a network participant decides not to cooperate. Since cooperation is fundamental for distributed environments such as mesh networks, misbehaviour can be a serious threat to them. In this work, the authors focus on the IEEE 802.11 EDCA medium access function which provides QoS in mesh networks. Simulation studies have been performed to determine what realistic forms of misbehaviour can occur and what their impact is. From these results the most beneficial forms of MAC layer misbehaviour in multihop mesh networks are derived.

Keywords: Mesh networks, QoS, IEEE 802.11, EDCA, misbehaviour

1 Introduction

Wireless mesh networks are steadily becoming a popular approach for providing network access to people's homes, especially in suburban and rural environments. Mesh networks allow a neighbourhood to share a single Internet connection, thus solving the last mile problem. They can also bring a community together by enabling easy and reliable data exchange within the network. By utilizing the latest technology, multimedia content can be exchanged over these networks.

Fig. 1 presents an aerial view of a mesh network. Each house in this neighbourhood has a wireless router, also called a Mesh Node (MN). These MNs form a backbone mesh network to provide robust connectivity. A mesh network can therefore be thought of as an immobile ad-hoc network. One of the MNs in the figure has a connection to the Internet and serves as a gateway for the other MNs. The MNs provide network access in each home. Wireless Access Points (APs) can be attached to the MNs to provide wireless access to household devices such as laptops, PDAs, tablet PCs, etc. The MN together with the AP is called the Mesh Point (MP). Stationary PCs can be directly connected to the MNs through Ethernet links.

The IEEE 802.11 standard [1] can provide wireless connectivity throughout the mesh network. It is currently the best choice when building a mesh network, because 802.11 equipment has become popular, cheap, reliable, and secure. The MNs in the network can communicate with each other using the 5 GHz frequency band and the user devices can connect with the APs using the 2.4 GHz frequency band. This makes the community-wide mesh part of the network separate from the wireless network in each household. The Enhanced Distributed Channel Access (EDCA) function ensures Quality of Service (QoS) at the Medium Access Control (MAC) layer and facilitates the exchange of multimedia content over the network. It provides traffic prioritization with four Access Categories (ACs) to provide appropriate QoS. These categories are, from the highest priority: Voice (Vo), Video (Vi), Best effort (BE), and Background (BK). In the upcoming 802.11 standard for mesh topologies – 802.11s [2] – EDCA is included as a mechanism for providing QoS. Therefore, EDCA is the main focus of the research presented in this paper.

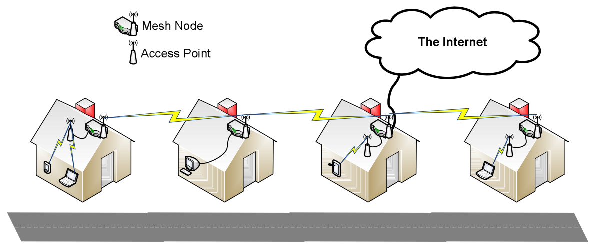

Fig. 1. Mesh network

Mesh networks rely on the cooperation of all participants. A problem arises if one of the participants misbehaves (i.e., decides not to cooperate with others). A mesh node may decide to misbehave in order to gain certain measurable profits (such as higher throughput). Misbehaviour is always done at the cost of the well-behaved nodes in the network. Therefore, it would be favourable if such actions were at least discouraged, if not made impossible.

Misbehaviour is a threat to networks built with the 802.11 standard because it provides no incentives to cooperate. Medium access in 802.11 is based on CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) and a set of pre-defined parameters. In EDCA, each AC has its own set of parameters: AIFS (Arbitration InterFrame Space), CWmin and CWmax, and TXOP (Transmission Opportunity) (Table 1).

Table 1. Values of EDCA Parameters

AC	AIFS	CWmin	CWmax	TXOP [μs]
Voice	2	7	15	3264
Video	2	15	31	6016
Best effort	3	31	1023	0
Background	7	31	1023	0

Any user can change these parameters to his/her own advantage. This can be done very easily with the use of the latest wireless drivers [3]. With these modifications, users can, for example, achieve better network access than their neighbours. Likewise, a vendor of wireless cards might decide on using non-standard parameters to achieve better performance. This makes misbehaviour a real threat to mesh networks. This problem has already been the subject of recent studies regarding cooperative environments such as mobile ad-hoc networks (Section 2). However, no research has been performed on the topic of providing QoS in misbehaviour-prone mesh networks.

Section 3 provides simulation results which determine the impact of misbehaviour on QoS provisioning in a multi-hop mesh environment. The focus of this work is on realistic misbehaviour, i.e., actions which are easy to perform and beneficial to the malicious user. The simulations consider modifying MAC layer parameters to either upgrade one's own traffic or to downgrade the traffic of others. These simulations show how beneficial different types of misbehaviour actually are. Finally, Section 4 concludes the paper and describes future work.

2 State of the Art

The problem of misbehaviour, especially in the context of mobile ad-hoc networks, has been the subject of study for the last several years. The first approaches to detecting misbehaviour were focused on the problem of not forwarding packets. Such actions are done at the IP layer and can be performed with the use of a firewall. The first benefit is that the misbehaving node has more bandwidth for its own traffic. Secondly, in the case of mobile nodes, it can extend its battery life.

The first solution to not forwarding packets was presented in [4] and later independently developed into CONFIDANT [5] and CORE [6]. This family of solutions is based on promiscuous observation of events in the network. Many types of misbehaviour can be detected, not only packets which are not forwarded, but also packet manipulation. Statistical algorithms are used to calculate a level of reputation for each node, which in turn determines cooperation. Misbehaving nodes (those with a low reputation) are gradually isolated from the network and thus such actions are discouraged.

The authors of [7] deal with the problem of MAC layer misbehaviour. They take into account several misbehaviour strategies, all dealing with manipulating the parameters of the contention window mechanism of 802.11. In their solution, it is the receiver, not the sender, which chooses the random backoff value. This value is transferred to the sender in either a CTS or ACK frame. Misbehaviour occurs when the sender deviates from that backoff.

Paper [8] presents DOMINO, an advanced software application designed to protect hotspots from greedy users. It monitors traffic, collects traces and analyzes them to find anomalies. DOMINO can detect many types of malicious and greedy behaviour, including backoff manipulation techniques. Anomaly detection is based on throughput (instead of observed backoff), which the authors acknowledge is not an optimal detection metric. The application can be seamlessly integrated with APs and it complies with standards. Additionally, a misbehaviour detection analysis in infrastructure-mode 802.11 EDCA WLANs can be found in [9]. However, both DOMINO and [9] cannot be used in distributed environments such as ad-hoc and mesh networks.

The authors of [13] present a simulation-based technique for detecting faults in wireless mesh networks. They utilize traces from a network monitor to perform simulations. The cause of the network behaviour can be detected, whether it is MAC layer misbehaviour, link congestion, or packet dropping. This is an interesting approach, however, it is not real-time and it depends on inaccurate simulations.

To summarize, there are several problems with the research efforts presented in this section. First of all, most research has been focused on WLANs operating in infrastructure mode. This is quite different from ad-hoc and mesh scenarios most notably because of the central access point. Secondly, the state of the art in misbehaviour detection is often focused on unrealistic misbehaviour. Examples include packet manipulation, selective jamming and other techniques which require expert skills. Also "adaptive" misbehaviour is considered, which is quite difficult to implement in real life. Furthermore, EDCA, with its four distinct sets of parameters, has not been taken into account in mesh network scenarios. Finally, the detection solutions are most often limited to only one layer of the OSI model (either Data Link or Network).

3 Analysis and Evaluation of Misbehaving Nodes

This section presents the results of an extensive simulation study of misbehaviour in mesh networks. The purpose of this analysis is to determine the impact that misbehaving users can have on QoS provisioning in such networks. All simulations were performed using the ns-2.28 simulator with a modified version of the TKN EDCA extension [10]. All the figures in this section present curves, where the error of each simulation point for a 95% confidence interval does not exceed 2% (this is too small for graphical representation).

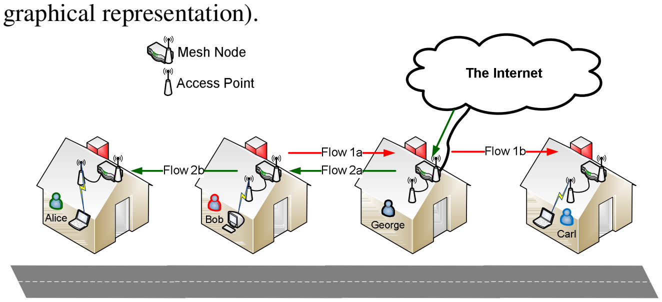

Fig. 2. Mesh network scenario

The simulated network topology is presented in Fig. 2. Each MN uses the EDCA function and is within range of its closest neighbour only. George's MN is a gateway to the Internet, Bob is sending a file to his friend Carl (Flow 1), and Alice is watching a video stream from the Internet (Flow 2). We can assume that UDP is used if Alice's transmission is real-time and TCP is used otherwise. Her traffic uses the highest priority (Vo) to ensure high quality of the video stream. If Bob uses a lower priority (BE) for his file transfer, the EDCA function will ensure that Alice's video stream is uninterrupted by Bob's file transfer. This is shown in the reference case (case A) in section 3.2. However, since Bob is in the path of Alice's traffic, he can misbehave by altering his medium access parameters. He can either simply degrade Alice's traffic (section 3.3) or combine this with promoting his own traffic (section 3.4). The question is: can such actions be beneficial for Bob? The answer is provided in section 3.5 which gives conclusions derived from the results of the simulations.

Since there is no impact of (and therefore no gain from) misbehaviour in non-saturated networks [11], we ensure that the simulated network is saturated. We evaluate the saturation throughput for the given topology in section 3.1. In saturation, the traffic source may not be relevant, so CBR was chosen. The packet size was 1000 B. In fact, the size of the packet is not that important because we are analyzing the behaviour of traffic priorities (and not absolute network performance). The RTS/CTS mechanism was not used since only Bob's and George's MNs generate traffic and they are neither hidden from, nor exposed to each other. The data rate of the simulated network was 11 Mbit/s and AODV was used as the routing protocol. The size of the network is small, but for one misbehaving node it is enough to show how its actions will influence network performance.

3.1 Saturation Throughput

In order to determine the saturation throughput of the network, the following simulation study was performed. The offered load of Flow 1 (Bob's file transfer) and Flow 2 (Alice's video stream) increased simultaneously from 64 kb/s to 12 Mb/s. The default priority (BE) was used for both flows. Both UDP and TCP were considered as the transport protocols. The results are presented in Fig. 3, which shows the average flow throughput achieved as a function of offered load.

For TCP the situation is clear – the saturation throughput is reached at approximately 1 Mb/s. This is the average end-to-end throughput of each flow. However, for UDP traffic, once a peak is reached, the throughput decreases to zero and congestion collapse occurs. This is because the interface queue present in the MAC layer of ns-2 uses the drop tail queue management algorithm. Bob's interface queue becomes completely filled with locally generated frames, leaving no room for frames that are to be forwarded. In real-life wireless cards such behaviour depends on the implementation. This does not occur for TCP traffic because this protocol adjusts its transmission speed using the additive increase/multiplicative-decrease algorithm. With respect to these results, an offered load of just over 2 Mb/s was chosen as the saturation throughput for this network scenario. In the following subsections, several different simulations were performed. Table 2 contains a brief description of all the considered cases.

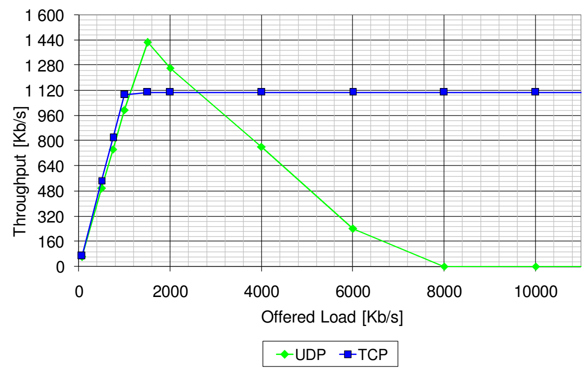

Fig. 3. Average flow throughput

Table 2. Descriptions of all cases

Case	Description
A	Reference situation, no misbehaviour
B	Bob changes the Vo parameters in his router to resemble BK priority (simple misbehaviour)
C	Case B + CWmin of forwarded traffic is set to maximal value (1023)
D	Bob uses CWmin = CWmax = 1 and TXOP = 8160 μs for his traffic
E	Case D + Case B (simple misbehaviour, change of forwarded traffic priority)
F	Case E + CWmin of forwarded traffic set to maximal value (1023)
G	Case F + Bob uses AIFS = 1

3.2 Reference Case

Case A is the reference situation. Alice uses Vo priority, whereas Bob consecutively uses each of the four EDCA priorities for his file transfer. Table 3 shows the throughput results that both flows achieved in the first (Flow 1a, 2a) and the second (Flow 1b, 2b) hop. Fig. 4 presents the end-to-end throughput values for both flows. If Bob is using the same priority as Alice (i.e., Vo) they both achieve similar throughput. Otherwise, if Bob uses a lower priority, his throughput is likewise lower. This is in accordance with the EDCA function. An interesting observation is that the decrease in throughput when Bob changes priorities from Vo to Vi is much larger for TCP than UDP. The explanation of this is that Flow 1 had to contend twice for the medium and twice with a lower priority. TCP is more sensitive than UDP to congestion, especially in wireless environments.

Table 3. Per-hop throughput results for case A (in Kb/s)

Flow 1 priority	UDP				TCP			
	F1a	F1b	F2a	F2b	F1a	F1b	F2a	F2b
Vo	1771	1364	1775	1363	1158	1096	1115	1055
Vi	1199	923	2111	2111	198	187	2039	1929
BE	1131	870	2111	2111	137	131	2095	1982
BK	1054	775	2111	2111	25	23	2207	2089

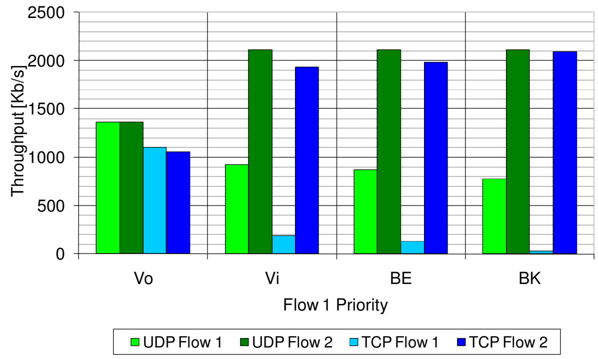

Fig. 4. End-to-end throughput results for case A

3.3 Downgrading Forwarded Traffic

In case B we assume that Bob runs a simple yet malicious script (perhaps found on the Internet) on his wireless router. This script changes the Vo parameters in his router to resemble BK priority. The priority of Alice's traffic is lowered but the frames are not manipulated. Again, Alice uses Vo priority, whereas Bob consecutively uses each of the four EDCA priorities for his file transfer. The throughput results (Fig. 5) again reveal interesting observations. When Bob is using Vo priority he sends his traffic using his modified EDCA parameters. This means that on the first hop, his traffic is sent at BK priority, and then forwarded as Vo priority (Fig. 6). For Alice's traffic, the priorities are reversed (first hop with Vo, second with BK). Why is Bob's end-to-end throughput higher? If we look at the hop-by-hop UDP throughput for Vo priority (Table 4) we see a similar situation as before: 100% of Bob's traffic and only 33% of Alice's traffic is forwarded. Again, locally generated traffic wins with traffic that is to be forwarded. When Bob uses Vi or BE priority he achieves the throughput gain that he was expecting. This gain is obviously higher for Vi than for BE. When Bob's file transfer is using BK priority, another interesting situation occurs. The per-hop use of priorities is shown in Fig. 7. When UDP is used, Alice's flow has more throughput (because it first has Vo and then BK whereas Bob's flow always has BK). However for TCP this is not the case, even though both flows have about 95% of traffic forwarded. This seems to be a similar case to the one described in [12], where it was shown that TCP may completely change throughput allocation independently of the EDCA configuration.

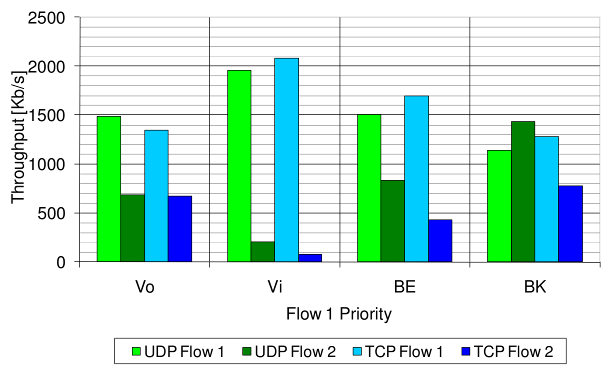

Fig. 5. End-to-end throughput results for case B

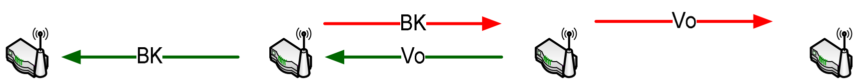

Fig. 6. Priorities used in Case B, Flow 1 priority: Vo

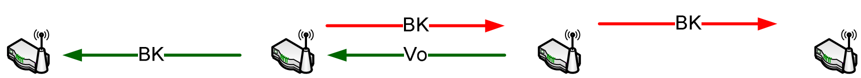

Fig. 7. Priorities used in Case B, Flow 1 priority: BK

Table 4. Per-hop throughput results for case B (in Kb/s)

Flow 1 priority	UDP				TCP			
	F1a	F1b	F2a	F2b	F1a	F1b	F2a	F2b
Vo	1482	1482	2111	686	1420	1343	710	672
Vi	1962	1958	2111	201	2192	2074	82	78
BE	1616	1503	2111	828	1787	1691	450	425
BK	1180	1136	2111	1430	1352	1279	815	771

Case C is similar to the previous one: Bob again modifies the Vo parameters in his router. This time he increases the CWmin parameter to its maximum value (1023). Bob has now degraded the Vo priority almost as severely as possible using EDCA parameter modification. The results are presented in Fig. 8 and Table 5. When Bob uses the Vo priority for his traffic, the situation is similar to that in case B. However, in this case the throughput values are significantly lower because of the high CW parameters. For all other priorities (Vi, BE, and BK) it can be seen that misbehaviour brings meaningful gains. The fact that Bob's throughput is high even if he uses BK signifies the importance of the CW parameters on throughput.

Table 5. Per-hop throughput results for case C (in Kb/s)

Flow 1 priority	UDP				TCP			
	F1a	F1b	F2a	F2b	F1a	F1b	F2a	F2b
Vo	428	428	2115	120	336	318	168	159
Vi	2052	2049	2111	7	2269	2147	3	2
BE	1917	1905	2111	34	2199	2081	19	18
BK	1775	1774	2111	57	2092	1980	34	32

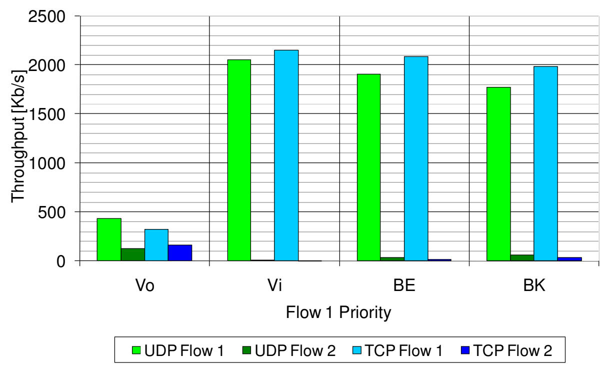

Fig. 8. End-to-end throughput results for case C

3.4 Promoting Local Traffic

In section 3.3 (cases B and C) Bob was gaining throughput by degrading the traffic parameters of forwarded traffic. In the following cases (D to G) we assume that Bob further manipulates EDCA parameters, this time in order to increase the medium access probability for his own traffic. In these cases Bob always uses the Vi priority for his file transfer. The results are presented in Table 6 and Fig. 9. In case D Bob uses the lowest possible CW parameters (CWmin = CWmax = 1) and the highest possible TXOP value (8160 μs). It might seem surprising that these parameters do not allow Bob to have a higher throughput than Alice. With UDP, he is able to achieve maximum throughput, but only on the first hop (Table 6). On the second hop this throughput decreases because Bob is using Vi priority, and Alice's traffic is using Vo priority. The results for TCP are similar, taking into account congestion control. In case E, Bob not only uses the most optimal EDCA parameters for Vi (like in case D) but also uses the simple misbehaviour that was presented in case B. This time, misbehaviour is advantageous for Bob in terms of achieved throughput. Case F differs from the previous one in that the CWmin parameter of Vo is increased to its maximal value (1023). The result is an even higher throughput for Bob. Finally, case G was modified from the previous one by also cheating on the AIFS value and changing it from 2 to 1. This brought a further, though minor increase in throughput.

Table 6. Per-hop throughput results for cases D, E, F, and G (in Kb/s)

Case	UDP				TCP			
	F1a	F1b	F2a	F2b	F1a	F1b	F2a	F2b
D	2111	662	1754	1755	260	246	1979	1873
E	2111	1878	2111	229	2191	2074	113	107
F	2111	2060	2111	32	2279	2157	20	19
G	2111	2111	2111	35	2318	2194	29	27

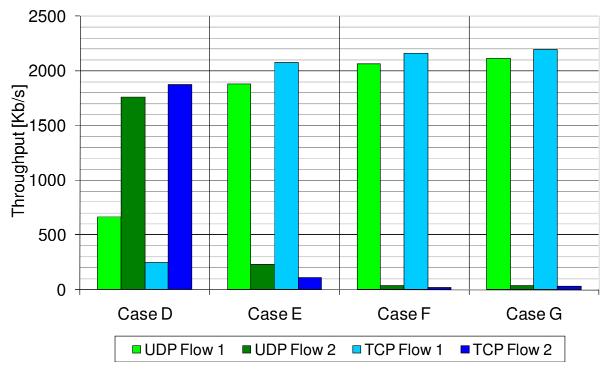

Fig. 9. End-to-end throughput results for cases D, E, F, and G

3.5 Lessons Learned

The results from the simulations have been gathered in Fig. 10, which presents the throughput gain that a misbehaving user can achieve. The gain was calculated as the ratio of the highest throughput in each case to the throughput achieved in case A (for Vo priority). Since the network was in saturation, it can be assumed that the gain of misbehaving Bob was equal to the loss of well-behaving Alice.

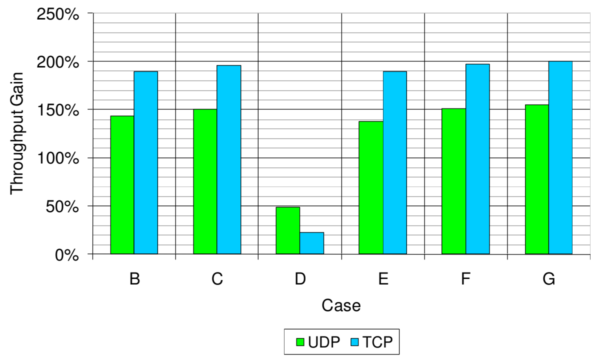

Fig. 10. Maximum throughput gain for misbehaving user

With the exception of case D, all the combinations of misbehaviour turned out to be very beneficial. For UDP there was a 40-50% increase, and for TCP – a 90-100% increase in throughput. The conclusion is that in all cases when Bob degraded the EDCA parameters of Alice's traffic he was able to achieve substantially higher throughput. He achieved best performance in case G, in which he both downgraded Alice's Vo traffic and promoted his Vi traffic. He changed his Vo priority parameters to resemble BK and additionally changed the CWmin of Vo to its maximum possible value. At the same time he changed the parameters of his Vi traffic to be optimal (i.e., lowest possible CWmin, highest possible TXOP, and lowest possible AIFS).

The unexpected result from these simulations is that, to achieve higher throughput in a multihop environment, it is significantly more important to degrade forwarded traffic than promote one's own. This problem has not been noticed before in literature and will influence future misbehaviour detection schemes. In multihop, EDCA-based networks, it is important to check for anomalies in the EDCA parameters used by neighbouring nodes. However, previous detection schemes focused only on detecting lowered parameters. The above results show that it is also necessary to monitor increased parameters, as this may lead to the downgrading of forwarded traffic.

4 Summary and Future Work

Misbehaviour occurs when a malicious user changes the settings of his/her MN in order to gain better medium access. This paper has presented the impact that realistic MAC layer misbehaviour has on QoS provisioning in mesh networks. Two forms of EDCA parameter modification were considered: downgrading forwarded traffic and promoting local traffic. It has been shown that this is a real threat to wireless mesh networks because it allows easy access to higher throughput and also degrades QoS provisioning. The main conclusion is that, in multihop scenarios, degrading forwarded traffic yields a greater advantage than cheating on medium access parameters.

Countermeasures to prevent misbehaviour are, therefore, required for mesh networks. Along this line, we envisage as future work the development of an architecture able to provide reliable multimedia content delivery, as well as, to deal with the problem of stations not adhering to standards. Based on the results presented in this paper, we will focus on detecting priority degradation of forwarded traffic. To this aim, an analytical model for detecting contention window manipulation in 802.11 EDCA mesh networks needs to be derived and some procedures to mitigate the influence of misbehaviour need to be proposed. These countermeasures should provide an incentive for the malicious users to cease their illegitimate actions.

Acknowledgement

The research leading to these results has received funding from the European Community's Sixth Framework Programme under grant agreement n° 0384239 (NoE CONTENT). The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 214994.

5 References

1. IEEE Standard for Information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999), pp. C1–1184, (2007)
2. IEEE, “802.11 TGs Mesh Networking” IEEE, Protocol Proposal IEEE P802.11s/D1.07, (2007)
3. MADWiFi – Multiband Atheros Driver for WiFi, http://madwifi-project.org
4. Kong J., Zerfos P., Luo H., Lu S., Zhang L.: Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks, IEEE ICNP (International Conference on Network Protocols) 2001, Riverside, (2001)
5. Buchegger S., Le Boudec J.Y.: Performance Analysis of the CONFIDANT Protocol: Cooperation of Nodes – Fairness In Dynamic Ad-Hoc Networks, In Proc. IEEE/ACM Symp. Mobile Ad Hoc Net. and Comp., Lausanne, Switzerland (2002)
6. Michiardi P., Molva R.: CORE: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks, Communication and Multimedia Security 2002, Portoroz, Slovenia, (2002)
7. Kyasanur P., Vaidya N.H.: Detection and Handling of MAC Layer Misbehavior in Wireless Networks, International Conference on Dependable Systems and Networks (DSN'03), p. 173 (2003)
8. Raya M., Hubaux J., Aad I.: DOMINO: a system to detect greedy behavior in IEEE 802.11 hotspots, Proceedings of the 2nd international Conference on Mobile Systems, Applications, and Services (MobiSys '04), Boston, MA, USA, (2004)
9. Serrano P., Banchs A., Kukielka J.F.: Detection of malicious parameter configurations in 802.11e EDCA, Global Telecommunications Conference 2005 (2005)
10. Wiethölter S., Emmelmann M., Hoene C., Wolisz A.: TKN EDCA Model for ns-2, Technical Report TKN-06-003, Telecommunication Networks Group, Technische Universität Berlin (2006)

11. Szott S., Natkaniec M., Canonico R., Pach, A.R.: Impact of Contention Window Cheating on Single-hop IEEE 802.11e MANETs. IEEE Wireless Communications and Networking Conference (WCNC 2008), Las Vegas (2008)
12. Banchs A., Azcorra A., García C., Cuevas R.: Applications and Challenges of the 802.11e EDCA Mechanism: An Experimental Study, IEEE Network, vol.19, no.4, pp. 52-58, (2005)
13. Qiu, L., Bahl, P., Rao, A., and Zhou, L.: Troubleshooting wireless mesh networks. SIGCOMM Comput. Commun. *Rev.*36, 5 (Oct. 2006)