Selfish Attacks in Two-hop IEEE 802.11 Relay Networks: Impact and Countermeasures

Szymon Szott and Jerzy Konorski

Abstract—In IEEE 802.11 networks, selfish stations can pursue a better quality of service (QoS) through selfish MAC-layer attacks. Such attacks are easy to perform, secure routing protocols do not prevent them, and their detection may be complex. Two-hop relay topologies allow a new angle of attack: a selfish relay can tamper with either source traffic, transit traffic, or both. We consider the applicability of selfish attacks and their variants in the two-hop relay topology, quantify their impact, and study defense measures.

Index Terms—Relay networks, IEEE 802.11, EDCA, QoS, game theory, selfish behavior, traffic remapping
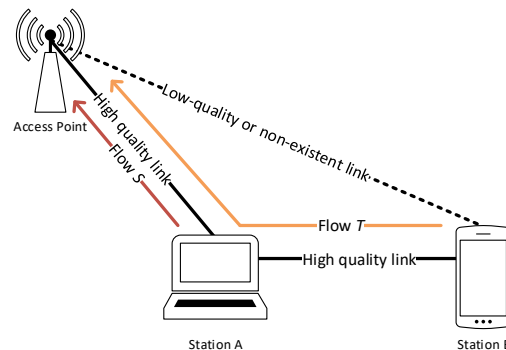

Fig. 1: Conceptual setting for selfish attacks in a two-hop relay network (all nodes with one wireless interface communicating on a single channel, neighboring nodes within carrier sensing range). Station A, as the relay, can increase its QoS perception by executing a selfish attack: upgrading source traffic (flow *S*), downgrading transit traffic (flow *T*), or both.

I. INTRODUCTION

THE coverage of IEEE 802.11 networks can be extended if stations connected to an access point (AP) act as relays, i.e., they share their connection with other, neighboring stations, who either cannot reach the AP themselves or have a poor direct connection to it. This approach creates a two-hop relay network (Fig. 1) which is known to have many advantages in terms of network coverage and performance [1].

A two-hop relay network requires cooperation from the relay station. If it declares cooperation, the relay may gain a privileged status at the AP, e.g., enjoying better terms of network access. In the following, we consider the specific case that the AP is a public hotspot and Internet access is granted free-of-charge provided that all one-hop connected stations act as relays to improve network performance[1]. In this setting, the relay may now be motivated to launch *selfish attacks* resulting in preferential treatment for its own (source) traffic over relayed (transit) traffic, hence to achieve an undue (and possibly undetected) increase of the quality of service (QoS) [2].

Selfish attacks can be launched in two ways. First, packet scheduling in the forwarding path can be biased in favor of source traffic. Second, source packets can be unduly prioritized at the MAC-layer. Since it is the MAC mechanisms that ultimately decide the order and delays of medium acquisition by successive packets (Section II), in what follows we focus on MAC-layer attacks defined in Section III.

Selfish MAC-layer attacks pose a serious threat to IEEE 802.11 networks (they are easy to perform, secure routing protocols do not prevent them, and their detection may be complex), but so far have been studied only in single-hop networks [2]. However, the two-hop relay topology adds another dimension to the problem, in that the relay can tamper with either source traffic, or transit traffic, or both (Fig. 1). We consider the applicability of these attacks and their variants in the two-hop relay topology (Section IV), quantify their impact (Section V), and study defense measures (Section VI). In our view, the presented attack scenario and proposed defence measures can serve as a useful analytical framework in any 2-hop configuration with a relay, in particular envisaged for 5G femtocells, regardless of the routing protocol and TRA method used.

II. QOS PROVISIONING IN IEEE 802.11 NETWORKS

QoS provisioning in IEEE 802.11 is achieved through the enhanced distributed channel access (EDCA) function, where higher-layer traffic classes are mapped to queues of one of four access categories (ACs). The ACs, in order of decreasing priority, are voice (VO), video (VI), best effort (BE), and background (BK). The configuration of each AC provides statistical prioritization with respect to channel access and duration [2].

Traffic classification into ACs is based on the Differentiated Services Code Point (DSCP) set in a packet's IP header: in the Type of Service (ToS) field in IPv4 or Traffic Class (TC) field in IPv6. DSCP values can be configured according to higher-layer policies, using network-layer packet mangling software (e.g., *iptables*) for all packets belonging to a given flow.

III. SELFISH ATTACKS IN IEEE 802.11 NETWORKS

The QoS provisioning model of IEEE 802.11 networks enables two types of selfish attacks: backoff attacks (BOAs) and traffic remapping attacks (TRAs). Both can be executed either as *source traffic upgrading* or *transit traffic downgrading* (Fig. 1), which we denote (⁺) and (⁻), respectively. We

S. Szott is with the Faculty of Computer Science, Electronics and Telecommunications, AGH University (e-mail: szott@kt.agh.edu.pl). His work is supported by contract no. 11.11.230.018.

J. Konorski is with the Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology (e-mail: jekon@eti.pg.gda.pl). His work is supported by his faculty's Statutory Fund.

[1]Technically, this requires that the relay act as a router, using either the path selection protocol of 802.11s or any ad hoc routing protocol.

consider only BOAs and TRAs because they are relatively easy to execute, in contrast to, e.g., forging frame headers [2].

A. Backoff Attacks

BOAs belong to the class of MAC parameter manipulation attacks – out of the available medium access parameters the contention window (CW) has proved to be the easiest to manipulate [3]. The CW governs the backoff mechanism, wherein each station *backs off* before accessing the channel by waiting a random number of time slots. An attacker may attempt to influence this random behavior so as to improve its QoS, either by decreasing CW for the AC of source traffic (BOA$^+$) or increasing it for transit traffic (BOA$^-$).

BOAs have two key advantages. First, modifying MAC parameters is often available as part a user's configuration interface. Second, detecting BOAs is challenging due to both the randomness inherent to the backoff function as well as the practical difficulties in performing precise time measurements.

The BOA$^+$ attack has mostly been studied in a single-hop infrastructure-based 802.11s setting and shown to effectively promote the attacker's traffic [4]. In a two-hop topology, a BOA$^-$ can additionally be applied to transit traffic to discriminate it in favor of source traffic. Research is required to quantify the impact of BOAs in two-hop relay settings.

B. Traffic Remapping Attacks

TRAs consist in claiming a different medium access priority through false DSCP settings so that traffic can be mapped onto a different AC. TRAs are simpler to perform than BOAs: a user application can access packet mangling software to change the current DSCP of any packet.

Upgrading TRAs (TRA$^+$) have been studied in single-hop ad hoc networks [2], where a distributed discouragement scheme, based on the threat of detection and punishment, allows TRAs only if they are harmless to honest stations; otherwise it induces selfish stations to learn that a long-sustained TRA is counterproductive. For relay networks, TRAs can be performed also on transit traffic to lower its priority (TRA$^-$). Such a possibility has thus far only been studied in a multi-hop ad hoc network [5], necessitating further research for the specific two-hop relay setting.

IV. TWO-HOP RELAY TOPOLOGY

We make the following assumptions for analyzing the network in Fig. 1. Station B and the AP are out of communication range. The placement of A in the topology allows the execution of any one of the previously discussed attacks. For ease of presentation we reduce the configuration space assuming that only two ACs are used: VO and BE. We consider the traffic transmitted in these ACs as a generic representation of high and low priority traffic, respectively. The interesting case for analysis is when, at A, the transit flow T is VO and the source flow S is BE. We evaluate the attack performance under saturation. We consider TCP because saturated UDP traffic causes a fairness problem – if a station's queue is full of source traffic then it drops all transit traffic. This is not the case for the self-regulating TCP.

TABLE I: Comparison of attacker's MAC-layer configuration for the BOAs and TRAs in the setting of Fig. 1. The last two columns indicate, respectively, which EDCA queue is used and what is its configuration. The latter denotes priority in medium access during frame transmission, the former – the QoS designation of the frames.

Attack strategy	Flow, intrinsic AC	AC queue used	Queue configured as
None (honest behavior)	T, VO	VO	VO
	S, BE	BE	BE
BOA$^+$	T, VO	VO	VO
	S, BE	BE	VO
BOA$^-$	T, VO	VO	BE
	S, BE	BE	BE
TRA$^+$	T, VO	VO	VO
	S, BE	VO	VO
TRA$^-$	T, VO	BE	BE
	S, BE	BE	BE
2xTRA (TRA$^+$ and TRA$^-$)	T, VO	BE	BE
	S, BE	VO	VO

A. Uplink and Downlink Scenarios

In the *uplink scenario*, there are two saturated TCP data flows terminating at the AP upon which A can execute the attacks: S, referred to as the source flow, and T, referred to as the transit flow. A's goal is to improve its uplink throughput (i.e., that of S). There are also traffic flows carrying TCP ACK segments in the reverse direction: S' and T' (omitted from Fig. 1 for clarity). These two flows are important: although they have a low rate, they impact the time spacing of the TCP data and thus end-to-end throughput of flows S and T.

A *downlink scenario* can also be considered, where the saturated TCP data flows are T and S'. Note that again A can directly influence T and S, the latter now consisting of TCP ACKs for flow S'. A's goal is now to improve its downlink throughput (i.e., that of S'). Whether MAC-layer attacks can have any impact on downlink throughput for two-hop relay settings is an open question that we want to address.

B. Attack Strategies

For each attack strategy it is helpful to specify which AC queues are used at the attacker, how they are configured, and which traffic is sent using which AC (Table I). For BOAs, we assume that AC queues are configured with valid EDCA parameters, e.g., in BOA$^+$ the BE queue is configured with VO parameters. Note that BOA$^+$ and TRA$^+$ differ in their effects even though they share the same configuration for S and T; similarly for BOA$^-$ and TRA$^-$. This is because under BOA$^+$ and BOA$^-$ both AC queues are used, whereas TRA$^+$ and TRA$^-$ merge source and transit traffic into a single AC queue, causing less inter-queue contention at the MAC layer. They also modify the QoS designation of each packet, which impacts its end-to-end as well as TCP ACK transmission priority, whereas BOA$^+$ and BOA$^-$ have local impact only.

BOAs and TRAs can also be combined into more sophisticated attack strategies. In particular, BOA$^+$ and BOA$^-$ (2xBOA) can be combined to produce a priority switch between the source and transit traffic flows (S and T). A similar

priority switch is produced by a combination of TRA$^+$ and TRA$^-$ (2xTRA).

In combinations of BOA and TRA, the TRA component assigns both flows to the same AC queue, for which the BOA component increases or decreases CW and other EDCA parameters. Thus the provided QoS is the same for both flows and only depends on the effectiveness of the underlying contention resolution. Hence, combinations of BOA and TRA come under performance optimization rather than network security, and as such are outside our scope. Henceforth we consider only 2xBOA and 2xTRA.

V. ATTACK IMPACT ANALYSIS

To study the impact of the considered attacks we used the ns-2.28 simulator. A multi-hop IEEE 802.11b[2] network served the three stations in Fig. 1. Important simulation settings include: enabling Request to Send/Clear to Send, setting transmission and interference range at one and two hops, respectively, using ad hoc on-demand distance vector routing, generating 1000-byte data packets, and using transmission queues of 50 packets. Simulations lasted 90 s (with a 15 s warm-up). Each data point is the mean of 20 experiment runs.

The provided QoS is defined based on a flow's intrinsic AC: as packet delay for VO traffic, i.e., T (which according to ITU-T recommendations should not exceed 100 ms), and as achieved throughput for BE traffic, i.e., S or S' in the uplink or downlink scenario, respectively. The selfish relay A attempts to maximize the throughput of S or S' at the cost of the victim flow T while maintaining a low risk of detection. We have simulated all attack strategies available to A (Table I). We have omitted from the figures attack strategies combining BOA and TRA (indicated above as out of scope), as well as 2xBOA, which was found only a marginal improvement over stand-alone BOA$^-$ (the contribution of BOA$^+$ turned out to be negligible).

A. Uplink Scenario

In the uplink scenario, from A's perspective, significant throughput gains of S can be achieved by attacking (Fig. 2a). However, it is visible that BOA$^+$ (where A uses both BE and VO queues configured with VO parameters) is not beneficial in two-hop relay settings. The reason is that, in addition to the inter-queue contention that worsens the overall performance, the VO queue used by T retains its relative priority: the TCP ACKs for T are handled as VO traffic at the AP, creating a smaller round-trip time than that experienced by S (whose TCP ACKs are handled as BE traffic). Under TRA$^+$, flow S approaches the throughput achieved by T with A's honest behavior (the reference line in Fig. 2b). The downgrading attacks provide even better gains, which supports the hypothesis that upgrading source traffic is less important than downgrading transit traffic and elimination of inter-queue contention (note that compared to BOA$^-$, TRA$^-$ yields slightly better gains because it causes the AP to handle TCP ACKs for T as BE traffic). However, 2xTRA is strikingly beneficial.

---

[2]Simulations were also performed for newer IEEE 802.11 physical layers with the same qualitative results.

The throughput gains of S are accompanied by the loss in throughput by T (Fig. 2b). In most cases the loss ranges between 30% and 50% with the exception of 2xTRA, where 90% of the throughput is lost. Delay in all but the last case is below the ITU-T requirement of 100 ms (Fig. 2c). This shows that some selfish attacks can be harmless (inflict no QoS degradation for T), even in saturation conditions.

B. Downlink Scenario

In the downlink scenario both BOA$^+$ and TRA$^+$ provide a small (3–5%) throughput gain for S' by increasing the sending rate of the corresponding TCP ACKs at the price of increased collision rate due to smaller CW values (Fig. 2d). BOA$^-$ and TRA$^-$ reduce the collision rate due to larger CW values, therefore are much more beneficial for A; BOA$^-$ slightly less so because, cf. Section V-A, the VO queue used by T still has relative priority despite being configured with BE parameters, whereas TRA$^-$ eliminates the inter-queue contention at A. Unexpectedly for A, 2xTRA performs no better than TRA$^-$. The reason is that the TRA$^+$ component has TCP ACKs for flow S' handled as VO traffic at A. Thus S' experiences a lower round-trip time and its transmit window is excessively expanded; the resulting increased collisions between AP and relay transmissions ultimately lower the throughput of S'. In all considered downlink cases, T's delay, though sometimes elevated beyond the reference value (representing A's honest behavior), meets the ITU-T requirement of 100 ms (Fig. 2f).

VI. DEFENSE MEASURES

Once detected [2], [3], selfish MAC-layer attacks launched by the relay station A cannot be simply punished by banning the attacker from further communication (by deauthentication and blacklisting), as this would mean loss of network access for station B. Revoking A's privileged status at the AP is one possibility. However, a more convenient, self-regulatory approach would be to provide incentives for A's honest co-operation. For example, if the AP suspects an ongoing attack launched by A, it can drop ACK frames for its source packets or shape its source traffic; both these punishment measures can be considered a subtle form of denial of service. We evaluate them in the topology of Fig. 1 in the uplink scenario to show that they involve only a small computational overhead on behalf of the punisher AP and no transmission overhead at all.

The first measure, dropping ACK frames [4], has the punisher refrain from sending MAC-layer ACK frames for correctly received data frames but only those belonging to the attacker's flow (S in the uplink scenario and S' in the downlink scenario). The degree of penalty can be scaled by acknowledging an $\alpha \in [0, 1]$ portion of frames.

The second measure, traffic shaping, has the punisher apply traffic control to TCP flows related to the attacker's traffic (S in the uplink and S' in the downlink scenario), e.g., in the form of a leaky bucket filter with a controlled output rate. This rate can be proportional, by $\alpha \in [0, 1]$, to the attacker's rate during an attack, so that for $\alpha = 1$ the throughput of S is equal to that shown in Fig. 2a.
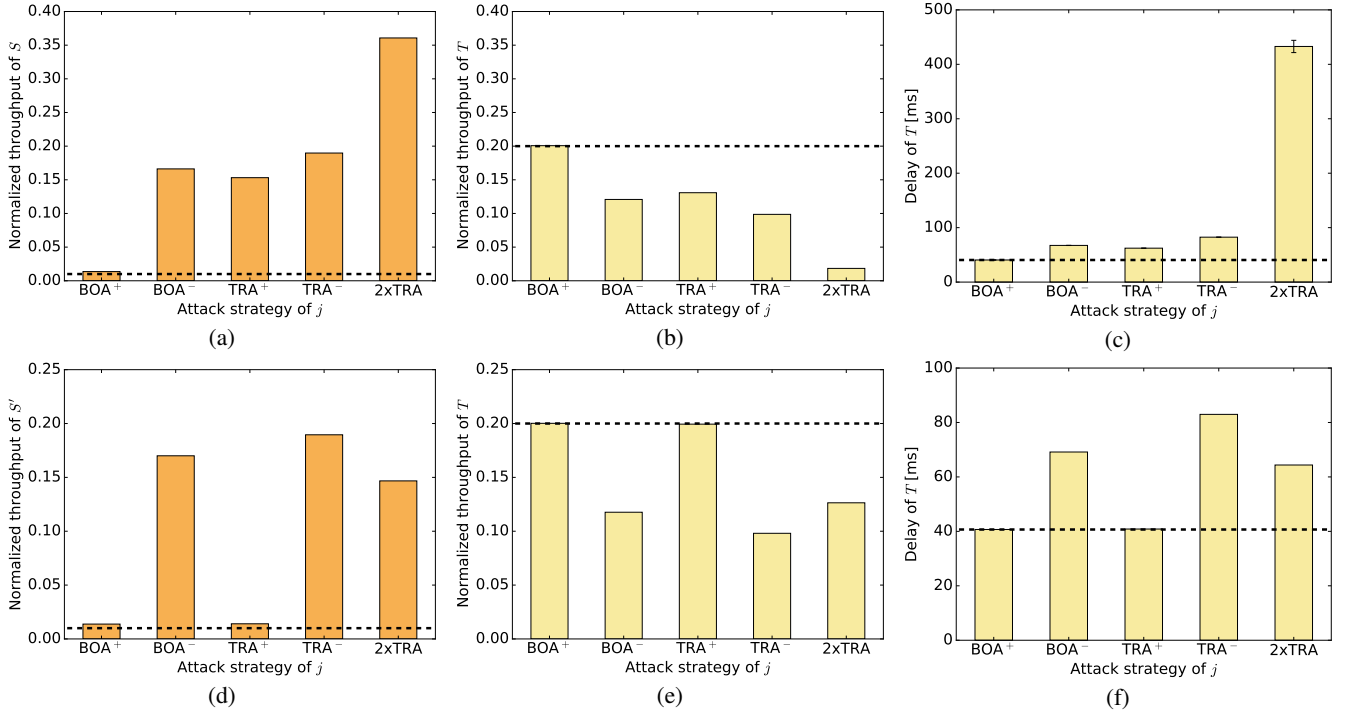
Fig. 2: Impact of A's attack strategy in the uplink (top) and downlink (bottom) scenario of the Fig. 1 topology: throughput of S/S' (left), throughput of T (middle), and delay of T (right). Throughput is normalized to the PHY data rate. 95% confidence intervals are either shown or too small for graphical representation. The reference line is for the case of A's honest behavior.
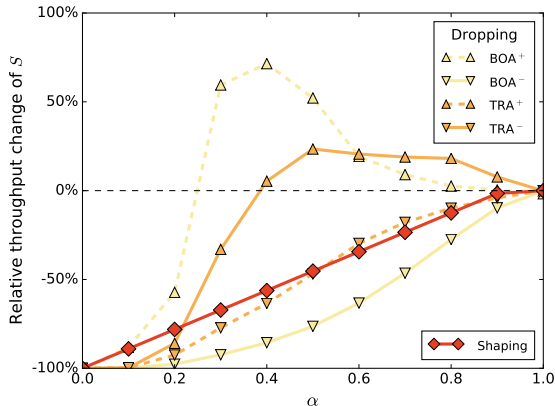

Fig. 3: Change of throughput of S caused by punishment at the AP in comparison to unpunished attack.

The results presented in Fig. 3 show that ACK dropping was able to scale S's throughput for the downgrading attacks. However, for the upgrading attacks, dropping ACKs can inadvertently optimize S's TCP flow with respect to the effect of hidden stations[3] and thus cause an unexpected *increase* of S's throughput for $\alpha \in [0.4, 0.9]$. We have also evaluated the performance of flow T (not shown) and observed that for TRA$^-$ the throughput is reduced for both S and T; clearly, ACK dropping has caused *network* performance degradation. In contrast, traffic shaping allowed to selectively (and almost linearly) control the throughput of S. Similar results (not presented here) were obtained in the downlink scenarios.

---

[3]Careful pacing of TCP transmissions could improve flow S's performance in the presence of hidden stations even without ACK dropping, although it might necessitate cross-layer optimization [6].

VII. CONCLUSIONS

We conclude the following: 1) except for 2xTRA in the uplink scenario, station A's selfish attacks were found harmless, since they did not violate the QoS requirements of high-priority traffic, 2) unlike in single-hop networks, BOA$^+$ brings the attacker no benefit in two-hop relay networks because it does not modify the packet's QoS designation, 3) downgrading attacks outperform their upgrading counterparts (particularly in the downlink scenario), while combined attack strategies were found to be only beneficial in one case (2xTRA in uplink), 4) simple ACK dropping is not a valid punishment in two-hop relay networks because of its unpredictable behavior; certainly, more sophisticated ACK dropping needs to be studied in the future. Future work should also consider more advanced incentive methods for relays, practical evaluation of the proposed defense methods both under TCP and UDP traffic, and analysis of new attack vectors enabled by 802.11ac/ax.

REFERENCES

[1] A. Garcia-Saavedra, B. Rengarajan, P. Serrano, D. Camps-Mur, and X. Costa-Perez, "SOLOR: Self-optimizing WLANs with legacy-compatible opportunistic relays," *IEEE/ACM Transactions on Networking*, vol. 23, no. 4, pp. 1202–1215, 2015.

[2] J. Konorski and S. Szott, "Discouraging traffic remapping attacks in local ad hoc networks," *IEEE Transactions on Wireless Communications*, vol. 13, pp. 3752–3767, 2014.

[3] P. Patras, H. Feghhi, D. Malone, and D. J. Leith, "Policing 802.11 MAC Misbehaviours," *IEEE Transactions on Mobile Computing*, vol. 15, no. 7, pp. 1728–1742, 2016.

[4] M. Cagalj, S. Ganeriwal, I. Aad, and J.-P. Hubaux, "On selfish behavior in CSMA/CA networks," in *Proc. of INFOCOM*, 2005.

[5] J. Konorski and S. Szott, "Modeling a traffic remapping attack game in a multi-hop ad hoc network," in *Proc. of IEEE GLOBECOM*, 2017.

[6] H. Xie and A. Boukerche, "TCP-CC: cross-layer TCP pacing protocol by contention control on wireless networks," *Wireless Networks*, vol. 21, no. 4, pp. 1061–1078, 2015.