Post- print of: S. Szott and J. Konorski, "Traffic Remapping Attacks in Ad Hoc Networks," in IEEE Communications Magazine, vol. 56, no. 4, pp. 218-224, April 2018, doi: 10.1109/MCOM.2018.1700061.

Traffic Remapping Attacks in Ad Hoc Networks

Szymon Szott, *Senior Member, IEEE,* and Jerzy Konorski

***Abstract*—Ad hoc networks rely on the mutual cooperation of stations. As such, they are susceptible to selfish attacks which abuse network mechanisms. Class-based Quality of Service (QoS) provisioning mechanisms, such as the enhanced distributed channel access (EDCA) function of IEEE 802.11, are particularly prone to traffic remapping attacks (TRAs), which may bring an attacker better QoS without exposing it to easy detection. Such attacks have been studied in wireless LANs, whereas their impact in multi-hop settings is less known. We provide a definition of TRAs highlighting their ease of execution and analyze their impact in multi-hop networks. Suitable detection methods and defense measures are also discussed and evaluated.**

***Index Terms*—ad hoc network, IEEE 802.11, EDCA, QoS, selfish behavior, traffic remapping.**

Introduction

To ensure correct provisioning of network services, wireless ad hoc networks rely on inter-station cooperation such as collective path discovery and forwarding of each other's traffic. This cooperation reduces a station's resources available for source packets, since transmission of routing advertisements or transit packets consumes energy and channel bandwidth. It also makes various kinds of attacks possible. *Malicious* attacks aim to destabilize the network's operation; examples are *rerouting* (issuing incorrect routing advertisements) and *blackhole*/*grayhole* (claiming the shortest path and dropping all or selected packets). Such attacks have led to the development of secure routing protocols [1]. In contrast, *selfish* attacker stations abuse network mechanisms to achieve an undue increase of the quality of service (QoS) or resource savings. For example, a selfish variety of the rerouting attack permits a station to prevent establishing paths that traverse it, hence shirk from forwarding transit packets. The *packet dropping attack*, i.e., refusing to forward offered transit packets, permits the station to achieve the same goal even if traversing paths have been established. Such attacks can be addressed by trust management frameworks [2], [3], whereby an exchange of observations of stations' past (mis)behavior is used for identification of selfish stations and their exclusion from the path discovery process.

Selfish rerouting or packet dropping attacks need not improve the QoS of the attacker station's source traffic, since the offered transit traffic may still be high on account of, e.g., transport-layer retransmissions of undelivered packers, and/or more interference from neighbor stations. More effective promotion of source traffic can be achieved through MAC-layer attacks targeting certain popular MAC protocols such as IEEE 802.11. A straightforward manipulation of medium access parameters responsible for transmission deferment (in particular the contention window, as in the *backoff* attack [4], [5]) disturbs the order of medium acquisition causing source packets to be unduly prioritized [6]. However, such attacks require tampering with the wireless card drivers, can be detected by a neighbor station carefully following the timing of transmissions sensed in the radio channel, and their impact is one-hop, since MAC parameters only have local significance.

S. Szott is with AGH University, Poland. His work is supported by the Polish National Science Centre (decision no. DEC-2011/01/D/ST7/05166).

J. Konorski is with the Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology, Poland.

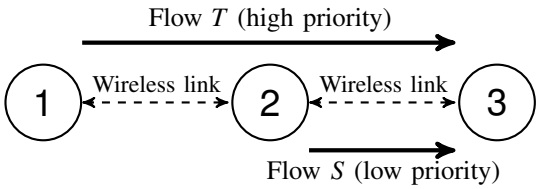

Fig. 1: Conceptual setting for a TRA. Stations 1 and 3 are out of hearability range. The attacker station 2 can increase its QoS perception by unduly promoting source traffic (low-priority flow S), demoting transit traffic (high-priority flow T), or both.

Contemporary wireless networks increasingly support QoS differentiation based on traffic classification, both at the routing layer using, e.g., the DSCP field in IP headers, and at the MAC layer using, e.g., the enhanced distributed channel access (EDCA) function of IEEE 802.11. While responding to the need for integrated data and real-time traffic handling over a common transmission substrate, this opens the way for a new class of selfish *traffic remapping* attacks (TRAs) [7]. Their mechanism is simple: the attacker station falsely assigns source packets to a traffic class mapped to high-priority MAC handling and/or transit packets to a traffic class mapped to low-priority MAC handling. Fig. 1 provides a conceptual setting for a TRA. Such attacks pose a serious danger to ad hoc networks for several reasons:

- like backoff attacks, they offer priority medium access to some traffic classes at the cost of others [7]; however, they do not require tampering with the wireless card drivers (MAC parameters remain unchanged) and may have multi-hop impact if the falsely assigned traffic class is honored by the on-path stations downstream of the attacker,
- like packet dropping attacks, they can be disguised as unintentional fault or temporary lack of resources; however, they directly promote source traffic (rather than only save resources for it), and are harder to detect (indeed, detection of undue promotion of source traffic requires costly deep packet inspection [7]), whereas undue demotion of transit traffic eludes packet delivery ratio-based detection and requires a properly incentivized watchdog mechanism [2] at a neighbor station),
- unlike packet dropping attacks, they cannot be properly

TABLE I: An example mapping between ITU-T Recommendation Y.1541 CoS, DiffServ PHB, and IEEE 802.11 EDCA ACs as derived from RFC 4594 and the `madwifi` Linux driver code. Also shown are the EDCA parameters ensuring per-AC QoS differentiation for the IEEE 802.11 HR/DSSS physical layer: minimum and maximum contention window (CW), arbitration inter-frame space number (AIFSN), and transmission opportunity (TXOP) limit.

CoS	DiffServ PHB	DSCP	AC	CWmin/CWmax	AIFSN [slots]	TXOPlimit
0	EF	`0x2E`	VO	7/15	2	3 ms
1	CS5	`0x28`	VI	15/31	2	1.5 ms
2, 3, 4	DF	`0x00`	BE	31/1023	3	single frame
5	CS1	`0x08`	BK	31/1023	7	single frame

addressed by trust management frameworks even if the detection difficulties are overcome: due to the multi-hop impact of TRA, large portions of paths rather than single misbehaving stations would have to be excluded from path discovery, and
- they can be launched flexibly as promotion of source traffic, or demotion of transit traffic, or both, thereby further impeding their detection.

Studies of TRAs in single-hop wireless LANs have shed some light on their effects and possible countermeasures (as explained in the next section). To date, no such study has been conducted in multi-hop ad hoc networks, where a few issues call for more attention. First, the impact range of a TRA is unclear, given the complex interplay of MAC contention, interference from hidden stations, and transport-layer flow control. This interplay also blurs the QoS perception and makes it difficult for honest (non-attacker) stations even to decide with certainty that a TRA is in progress, especially in the absence of single-broadcast hearability. The latter moreover rules out simple punitive measures such as threats of jamming [7], [8]; later we also show that certain other measures that work well in single-hop settings, such as ACK dropping, may fail in multi-hop settings. On the other hand, undue promotion of source traffic can potentially be neutralized by a demotion of transit traffic at a downstream station. We offer a simple simulation setting and a preliminary study of the multi-hop effects of TRAs, discuss possible detection methods as well as adopt some MAC- and transport-layer mechanisms to propose promising defense measures.

Related Work

Existing security research has mostly focused on preventing selfish manipulation of medium access parameters in wireless LANs, and malicious and resource-saving selfish routing attacks in multi-hop wireless networks. This is a consequence of recognizing MAC and routing as the key functionalities involved in single- and multi-hop transmissions, respectively. Consequently, TRAs have so far mostly been studied in single-hop wireless network settings. Li and Prabhakaran [9] proposed a coordinated QoS framework of admission control and priority reallocation assuming that devices are honest and independent of the local applications (users), which contradicts the usual conviction that a wireless station's equipment and users are united by a common goal. Ghazvini et al. [10] proposed a game-theoretic approach to set appropriate transmission opportunity values (a form of channel reservation) for stations to maximize individually weighted combinations of throughput and delay. The authors of [7], [8] proposed a distributed scheme based on the threat of detection and subsequent jamming; unless TRAs are harmless to honest stations, selfish stations learn that a long-sustained TRA is counterproductive. To the best of our knowledge, selfish TRAs have not been studied in multihop ad hoc networks, and so their impact remains unknown. In very general terms, a TRA can be considered a violation of access rights and dealt with using an IDS (intrusion detection system) framework [11], [12], though IDS solutions are mostly oriented towards malicious attacks; also, the underlying signature or anomaly detection methods are likely to fail against TRAs unless deep packet inspection is used. This scarcity of previous work and the above listed dangers of TRAs have motivated our research.

QoS Provisioning in Ad Hoc Networks

TRAs exploit the class-based philosophy of QoS provisioning in ad hoc networks, whereby a station performs traffic classification to decide how QoS is differentiated. The definitions of the underlying traffic classes vary by OSI layer and standardization body. ITU-T, in Recommendation Y.1541, defines Classes of Service (CoS). These are translated to a Differentiated Services (DiffServ) per-hop behavior (PHB). The CoS-to-PHB mapping is done at the source station according to administrator policies. Each PHB is associated with a Distributed Services Code Point (DSCP) set in its IP header (in the `Type of Service` field in IPv4 or `Traffic Class` field in IPv6).

At the MAC layer, IEEE 802.11 uses the enhanced distributed channel access (EDCA) function to provide QoS. In EDCA, the higher-layer traffic class is mapped to one of four defined access categories (ACs), in order of decreasing priority: voice (VO), video (VI), best effort (BE), and background (BK). Each category has its own set of medium access parameters (Table I), to determine the probability and duration of channel access.

The CoS-to-DSCP-to-AC mapping is a non-trivial issue. It encompasses recommendations from different standardization bodies (IETF and IEEE), which do not provide a one-to-one mapping. Moreover, the DSCP-to-AC mapping is vendor-specific. Table I illustrates a possible CoS-to-DSCP-to-AC mapping based on RFC 4594 and the source code of the `madwifi` Linux wireless card driver. In practice, the CoS-to-DSCP mapping is implemented as a rule set within network-layer packet mangling software (such as Linux `iptables`), which allows setting DSCPs for all packets belonging to a given flow. The DSCP-to-AC mapping is embedded in

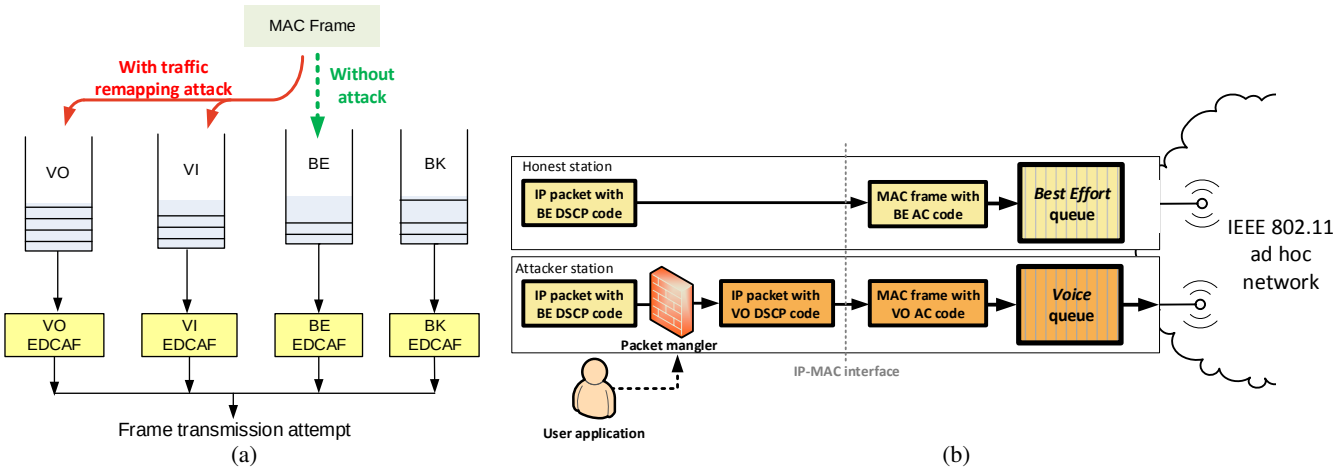

Fig. 2: TRA in an IEEE 802.11 EDCA setting: (a) MAC frames are classified into one of the four ACs and then handled by the appropriate EDCA function (EDCAF) to contend for medium access, (b) attack execution in an IEEE 802.11 ad hoc network [7].

the wireless card driver. As such it can be assumed to be unchangeable from an ad hoc network user's perspective.[1] Therefore, the packet mangling software in fact amounts to a CoS-to-AC mapping.

Traffic Remapping Attacks

A TRA in IEEE 802.11 EDCA networks consists in claiming a different medium access priority through false designation of the CoS so that it can be mapped onto a different AC (Fig. 2a). This attack is similar to the MAC parameter manipulation (backoff) attack in that it also increases medium access probability; however, it is much easier to perform: a user application issuing best-effort IP packets uses packet mangling software to replace the current DSCP with one corresponding to a higher CoS (Fig. 2b) without requiring access to the wireless card driver. Another advantage of TRAs is apparent in multi-hop networks: the attacker's packets are given more transmission opportunity along the whole downstream path instead of just in the attacker's single-hop vicinity. This is because the QoS designation of the packet is modified above the MAC layer and, unless downstream forwarders tamper with it, the packet will be considered high-AC at each hop. Furthermore, in multi-hop networks, an attacker may not only *promote* (upgrade the QoS of) source traffic as described above, but also *demote* (downgrade the QoS of) transit traffic. We denote by TRA$^+$ and TRA$^-$ the upgrading and downgrading variants of TRA, respectively. These variants can be combined at an attacker station to produce 2xTRA, a natural all-out attack strategy.

To understand how TRAs are executed, consider the setting of Fig. 1. This is the simplest multi-hop topology in which an attacker (station 2) can influence both source traffic (flow S) and transit traffic (flow T). For ease of presentation let only two ACs be used: VO and BE. The interesting case is when S is low-priority and T is high-priority. Thus the *intrinsic* AC (the AC to which its true CoS entitles) of S and T is BE and VO, respectively. For both TRA$^+$ and TRA$^-$, it is helpful to specify which AC queues are used at the attacker and which traffic uses which AC. Table II provides such specifications for standard IEEE 802.11 EDCA behavior as well as for each possible attack strategy of station 2: none (honest behavior), TRA$^+$, TRA$^-$, and 2xTRA (the latter performing a priority switch between the two flows). Note that TRAs modify the QoS designation of each packet (the 'Used AC' column), which impacts all its transmissions downstream of the attacker.

[1]Even if the user is able to tamper with the wireless card driver, tampering with the mangling software is easier and equally effective.

TABLE II: The attacker station's MAC-layer configuration for the TRA variants in the setting of Fig. 1. Note that TRA$^+$ only affects S and TRA$^-$ only affects T.

Attack strategy	Flow	Intrinsic AC	Used AC
None (honest behavior)	T	VO	VO
	S	BE	BE
TRA$^+$	T	VO	VO
	S	BE	VO
TRA$^-$	T	VO	BE
	S	BE	BE
2xTRA (TRA$^+$ and TRA$^-$)	T	VO	BE
	S	BE	VO

Attack Impact

Due to the complex interplay of EDCA contention, interference from hidden stations, intra-flow competition (packet transmissions from one station compete with those from up- and downstream stations), and TCP flow control, it is not clear whether selfish MAC-layer attacks always bring gains to the attackers and harm to other stations, in particular what impact they have over multiple hops. It is known that such attacks are most effective under heavy load [4]–[6]; on the other hand, end-to-end throughput is known to decrease with hop-length in ad hoc networks. Hence, even under saturation traffic, the source station of a long-path flow may not offer enough traffic

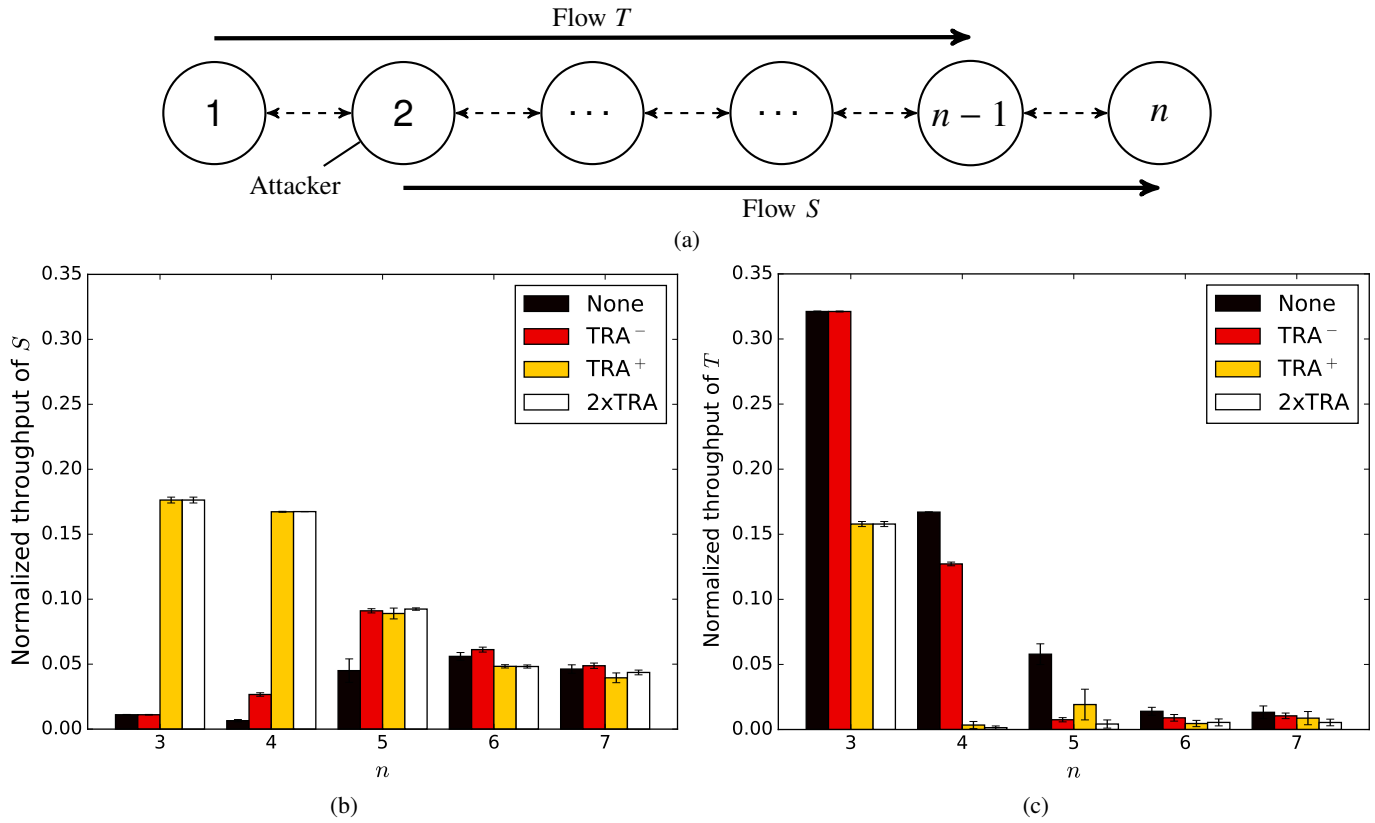

Fig. 3: Impact of TRAs for same-length paths of source and transit traffic. (a) Linear multi-hop topology with flows S and T sharing an $(n-2)$–hop subpath; (b), (c) throughputs of flows S and T normalized to the PHY data rate. 95% confidence intervals are either shown or too small for graphical representation.

to perform an effective TRA$^+$. Similarly, a TRA$^-$ performed on a long-path victim flow may be insignificant given that the victim flow's throughput is very low anyway. The above described attack strategies should be studied for many routing, traffic and station mobility scenarios, and various network topologies and sizes. However, to get initial qualitative insight as to whether and under what conditions selfish MAC-layer TRAs in multi-hop topologies are worth defending against, a simple small-size network is sufficient.

We set up a simulation model of a fixed linear multi-hop topology featuring a single attacker (Fig. 3a) and two packet flows. (Once this impact is visible, more general settings with multiple attackers need to be analyzed in the future.) In keeping with the above attacker model, both TRA$^+$ and TRA$^-$ must be meaningful; hence we assume that the attacker always lies on the victim flow's path and is itself a source of another flow. Furthermore, both flows use TCP and saturation conditions are maintained. Finally, although the victim flow's intrinsic AC is assumed to be VO, we record its end-to-end throughput rather than delay or packet loss ratio, as this yields more informative performance comparisons in our simple setting.

The linear topology in Fig. 3a features stations $1, 2, \ldots, n$, where station 2 is an attacker. As in Fig. 1, flows S and T move along paths directed from left to right.[2] The transit (victim) flow T originates at station 1 and terminates at station $n-1$. The source flow S originates at the attacker and terminates at station n. In the simulations, we vary n and thus the path lengths of S and T. This is intended to show the effect of the attacks when both flows' throughputs diminish as their common $(n-2)$-hop subpath gets longer.

All simulations were performed in ns-2 with the following experiment setup: 802.11 settings as listed in Table I, RTS/CTS enabled, transmission and interference range of one and two hops, respectively, AODV routing, and 1000-byte data packets generated at source.

The main conclusions from the results in Fig. 3b and 3c are:

- For up to two-hop flow paths ($n \leq 4$), the high-priority transit flow T has enough throughput to saturate the channel and significantly degrade the throughput of S when the attacker does not perform a TRA; with increased path length, T's throughput diminishes because of intra-flow competition and TCP flow control, and so leaves a larger bandwidth share to S.
- Since T suffers from increasing intra-flow competition as n increases beyond single-hop paths, 2xTRA brings visible gains to S and visible harm to T; both become

[2]TCP ACK flows moving in the opposite direction are omitted from the figure for clarity.

barely perceptible when, at large n, S itself starts suffering from intra-flow competition and TCP flow control.
- Obviously, TRA$^-$ cannot be applied when the paths are one-hop ($n = 3$), hence 2xTRA produces the same results as TRA$^+$ in this case.
- For two-hop paths ($n = 4$), TRA$^-$ brings some throughput gain for S and harm for T, but TRA$^+$ and 2xTRA differentiate the two flows' throughput much more distinctly: promoting S at the attacker station and the downstream station 3 outweighs the effect of weaker competition from a demoted T.
- For paths of four or more hops, TRA$^-$ is effective in terms of throughput gain for S and harm for T, the other TRAs being counterproductive: stronger intra-flow competition suffered by S outweighs the effect of its promotion at the attacker and all the downstream stations.

In other simulations we have observed that depending on the path characteristics and traffic scenario, TRAs (especially 2xTRA) may bring significant, sometimes huge gains to the attacker and harm to other stations. They are most felt within the range of a few hops and are more pronounced in the presented scenario than in one with the direction of flow S reversed. As the path length increases, the incentives to attack weaken on account of the diminishing throughput of high- and low-priority traffic alike. These facts make TRAs a serious security factor in multi-hop ad hoc networks, where path length is traded for station mobility to reduce interference and increase network capacity.

Attack Detection

Attack detection is the prerequisite of some defense measures against selfish attacks. For MAC-layer attacks, accurate attack detection is challenging. It can be done either *directly*, by noticing the attacker's deviations from standard behavior (through promiscuous observation of the radio channel), or *indirectly*, by observing the impact of the attacks on own or other stations' perceived QoS.

Directly detecting a parameter modification attack requires a watchdog mechanism to measure how many time slots an attacker uses for backoff; a statistical analysis is required to determine whether the observed samples fit into the desired distribution. Direct detection of TRAs determines whether the monitored higher-layer traffic matches its class designation, i.e., whether the structure of the user payload is characteristic of the traffic class that maps onto the AC specified in the MAC header. In particular, we need to recognize best-effort (e.g., HTTP, FTP, P2P) traffic being sent in the VO or VI ACs. A variety of traffic classification methods exist that serve either attack detection, as part of an IDS, or class-based QoS provisioning. Here, both these goals are addressed at once: TRAs are detected to eventually provide appropriate QoS to each traffic class. Depending on permissible detection time and available resources, the following traffic classification methods may be used, in order of increasing complexity:

- analysis of the transport protocol type and source and destination address/port is easy, but may fail against end-to-end encryption or non-standard port addresses,
- deep packet inspection, i.e., performing pattern matching on the packet payload requires high processing power, and may fail against end-to-end encryption or nonstandard protocols, and
- emerging machine learning [13] and data-mining techniques such as payload length analysis are efficient and insensitive to payload encryption[3]; they may require an offline learning period.

Note that flow monitoring for the detection of TRA$^+$ at a source station need only be performed by selected downstream stations and need not synchronize with the flows' lifetimes (any few packets 'testing positive' can identify a suspicious flow).

A watchdog can be used for detecting a TRA$^-$at a forwarder station. Referring to Fig. 1, a MAC-layer watchdog at station 1 will notice the priority change in the packet forwarded by the neighbor station 2 to the out-of-range station 3 provided that transmissions from 2 to 3 can be heard at 1. This requirement will not be met if multiple radio channels or directional antennas are in use, or if stations differ in their transmission range; the watchdog will also fail if transmissions between 2 and 1 are corrupted by collisions or noise, while those between 2 and 3 are not. Alternatively, a secure end-to-end signaling protocol can be applied to verify the priority of data packets delivered to the destination.

The aforementioned direct detection methods can be augmented with indirect detection. If a station notices that its QoS parameters fall below a pre-defined level then it can suspect a TRA is in progress. The station can then start a direct detection procedure or resort to the countermeasures described in the following section.

Finally, hybrid detection schemes can be employed. They usually involve trust management schemes [2], [3] whereby stations share the results of their direct or indirect attack detection.

Defense Measures

Within the existing defense approaches, TRAs in ad hoc networks can be made infeasible, mitigated, or disincentivized [6]. Under the first approach, both EDCA parameter configuration and flow-to-AC mapping are embedded in hardware or firmware as 'factory settings' and inaccessible at the wireless card driver level. In IEEE 802.11-based ad hoc networks it requires either tamper-proof hardware (which affects device cost and flexibility) or redesigned mechanisms (which affects compatibility with legacy installations).

Mitigating the impact of attacks, so that the QoS of honest stations is preserved, may be possible by isolating the attacker, e.g., by rerouting traffic over a different path. Isolation requires first detecting the attacker (which may be challenging, as discussed above) and then excluding it from prospective path discovery. Unfortunately, due to the multi-hop impact of TRAs, some or all stations downstream of the attacker may have to be excluded as well.

[3] Even with encrypted payload, the traffic class designation (e.g., DSCP) must be transmitted in the clear.

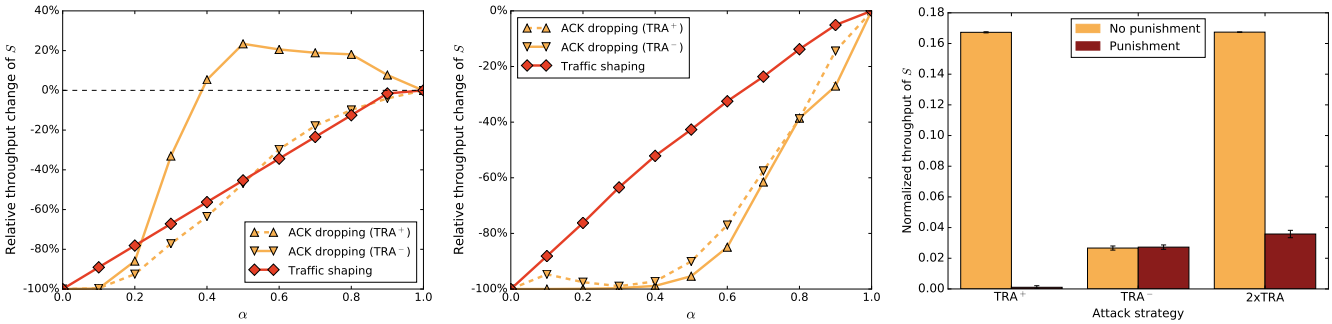

Fig. 4: Impact of attacker-aware (ACK dropping and traffic shaping) and attacker-unaware (punitive TRA$^-$) punishment on the throughput of S. For the former, throughput change relative to no-punishment case is presented for one-hop (a) and two-hop flow paths (b). For the latter, throughput normalized to the PHY data rate is presented for punishment and no-punishment cases (c).

Incentive schemes should be designed so as to teach the attacker that TRAs are not beneficial. This can be achieved through credits, threats, or punishment.

Credit-based incentive schemes have a station earn credits for participating in the network operation, e.g., for forwarding transit traffic. Credits can then be spent when using forwarding services of other stations. While this approach disincentivizes packet dropping attacks [14], it seems less applicable at the MAC layer. Placing a monetary value on traffic forwarding is straightforward (e.g., one packet earns one credit), however, charging for MAC-layer services based on the provided QoS would require a complex economic model, since it is not clear how the earnings could buy similar services of other stations.

A threat of punishment can disincentivize a selfish attacker if the punishment is serious enough (such as jamming the attacker's packets). In [7], disincentivizing a TRA in a single-hop setting is studied using game-theoretic methods. Stations suspecting the low QoS they are perceiving is caused by a TRA broadcast special messages to signal imminent punishment (e.g., jamming of attackers' data packets upon deep packet inspection). Honest stations can ignore such messages, while selfish attackers can choose between continuing and abandoning the attack. If they perceive the condition of imminent punishment as costly, a noncooperative game arises at whose Nash equilibrium TRAs become either counterproductive or harmless. A similar game-theoretic analysis, out of the scope of this paper but intended as future work, would provide specifics of disincentivizing TRAs in multi-hop networks.

Finally, actual punishment of an attack can be meted out at a *punisher* station. Punishment-based defense measures can be categorized as *attacker-aware* or *attacker-unaware*. The former target specific flows once the punisher identifies their sources as attackers. Some well-known possibilities include dropping ACK frames for the attacker's source packets, shaping the attacker's source traffic, or banning the attacker from further communication (by deauthentication and blacklisting). In contrast, attacker-unaware punishment is uniformly applied to all traffic so that the QoS of attackers is adversely affected, whereas that of honest stations is not. For example, a punitive TRA, similar to [9], may be effective if the punisher has reasons to believe that honest stations' flows are BE (hence, will remain unaffected). Likewise, existing congestion control mechanisms may be applied if honest stations are believed to generate a moderate traffic volume. All these methods require that the punisher be downstream of the attacker (e.g., station 3 in Fig. 1). Below we evaluate the performance of selected punishment-based defense measures.

Attacker-Aware Punishment

We propose two punishment-based measures that build a form of indirect control into the MAC- and TCP-ACK mechanisms. Both measures impose only a small computational overhead and no transmission overhead upon the punisher. We use the linear topology of Fig. 3a for either $n = 3$ (one-hop flows) or $n = 4$ (two-hop flows); station 3 is the punisher[4].

The first measure consists in dropping MAC-layer ACK frames, a well-known defense method for single-hop networks [4]. In multi-hop networks, it has the punisher refrain from acknowledging correctly received DATA frames of the attacker's source flow. The penalty can be calibrated by acknowledging an $\alpha \in [0, 1]$ portion of frames.

The second measure consists in traffic shaping, where the punisher applies traffic control to the attacker's TCP flows, e.g., a leaky bucket filter with a controlled output rate proportional, by $\alpha \in [0, 1]$, to the attacker's rate. This approach will work only for steady rate long-lived streams, which we assume.

We first evaluate the effectiveness of the attacker-aware punishment when the attacker's flow path is single-hop (Fig. 4a). The results show that ACK dropping is able to scale flow S's throughput under TRA$^-$. However, under TRA$^+$, ACK dropping can inadvertently optimize S's TCP operation in the presence of hidden stations [15] and thus cause an unexpected *increase* of its throughput for $\alpha \in [0.4, 0.9]$, while T suffers from a throughput loss (not shown). We conclude that ACK dropping may strengthen the undesirable QoS differentiation

[4]Technically, as the destination of the attacker's flow, station 3 has no incentive to punish the attacker. This part of the analysis is therefore used only to illustrate the potential effectiveness of the studied defense method.

caused by TRAs. In contrast, traffic shaping allows to effectively (nearly linearly) control the throughput of S.

We next extend the path length of S to two hops (Fig. 4b). Again, traffic shaping allows to effectively control the attacker's throughput, whereas ACK dropping ensures linear control for $\alpha \in [0.5, 1]$ and all but chokes S for smaller α values. We conclude that ACK dropping, while effective in single-hop networks, cannot be a valid punishment in multi-hop networks because of its unpredictable behavior.

Attacker-Unaware Punishment

By its nature, TRA$^-$ may serve as a punitive measure if performed on a flow on which a TRA$^+$ was previously performed. It is attacker-unaware in that it does not affect flows already configured as BE. We analyze its effects depending on the attacker's strategy in the topology of Fig. 3a for $n = 4$ (both S and T traverse two hops), the punisher being station 3. The attacker's throughput is affected only for TRA$^+$ and 2xTRA. For TRA$^-$ the punishment becomes irrelevant, as S is already configured as BE (cf. Table II). We conclude that a punitive TRA$^-$ is effective against TRAs whenever they are relevant, while not requiring attacker detection.

Conclusion

Traffic remapping attacks consist in modifying the medium access priority of flows to unduly promote source or demote transit traffic. The effects of local-scope manipulation by the attacker can translate into end-to-end per-flow symptoms. The studies presented here as well as other results in the literature regarding these attacks lead to the following conclusions. First, unlike in single-hop WLANs, selfish MAC-layer attacks in multi-hop topologies can sometimes be harmless (not necessarily degrade the QoS of honest stations' flows); if they do bring harm, however, they pose a serious threat and their detection is complex. Second, TRAs have a multi-hop impact, albeit mostly on short paths, whereas parameter manipulation attacks are local-scope and bring the attacker little or no gain in multi-hop wireless topologies. Third, TRA$^+$ creates the highest threat for multihop ad hoc networks by combining sizable gains with a low risk of detection. Finally, if attacker detection is available, punishment via traffic shaping is preferable to ACK dropping; otherwise a punitive TRA$^-$ is an option.

Based on these conclusions, future work regarding MAC-layer selfish insider attacks in ad hoc networks should focus on TRAs rather than MAC parameter manipulation attacks. We foresee analyzing more complex topologies and traffic patterns as well as the presence of multiple attackers. It remains to be seen how harmful, harmless, or perhaps sometimes beneficial TRAs are in these settings. A game-theoretic analysis of TRAs in a multi-hop setting might help design incentive-based defense measures.

Acknowledgment

The work of Szymon Szott is supported by the Polish National Science Centre (decision no. DEC-2011/01/D/ST7/05166). Jerzy Konorski developed some preliminary ideas of the paper during his participation in the Future Internet Engineering project supported by the European Regional Development Fund under Grant POIG.01.01.02-00-045/09-00.

References

[1] A. Anand, H. Aggarwal, and R. Rani, "Partially distributed dynamic model for secure and reliable routing in mobile ad hoc networks," *Journal of Communications and Networks*, vol. 18, no. 6, pp. 938–947, 2016.

[2] Z. Movahedi, Z. Hosseini, F. Bayan, and G. Pujolle, "Trust-distortion resistant trust management frameworks on mobile ad hoc networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 18, no. 2, pp. 1287–1309, 2016.

[3] S. Tan, X. Li, and Q. Dong, "A trust management system for securing data plane of ad-hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7579–7592, 2016.

[4] Y. woon Ahn, A. M. K. Cheng, J. Baek, and P. S. Fisher, "Detection and punishment of malicious wireless stations in IEEE 802.11e EDCA network," in *2010 IEEE Sarnoff Symposium*, 2010.

[5] J. Konorski, "A game-theoretic study of CSMA/CA under a backoff attack," *IEEE/ACM Transactions on Networking*, vol. 14, no. 6, pp. 1167–1178, 2006.

[6] S. Szott, "Selfish insider attacks in IEEE 802.11s wireless mesh networks," *IEEE Communications Magazine*, vol. 52, no. 6, pp. 227–233, 2014.

[7] J. Konorski and S. Szott, "Discouraging traffic remapping attacks in local ad hoc networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 7, pp. 3752–3767, 2014.

[8] ——, "Credibility of Threats to Jam Anonymous Traffic Remapping Attacks in Ad Hoc WLANs," *IEEE Communications Letters*, vol. 21, no. 3, pp. 624–627, 2017.

[9] M. Li and B. Prabhakaran, "MAC layer admission control and priority re-allocation for handling QoS guarantees in non-cooperative wireless LANs," *Springer Mobile Networks and Applications*, vol. 10, no. 6, pp. 947–959, 2005.

[10] M. Ghazvini and N. M. K. Jamshidi, "GTXOP: A Game Theoretic Approach for QoS Provisioning Using Transmission Opportunity Tuning," *PLoS ONE*, vol. 8, no. 5, 2013.

[11] E. M. Shakshuki, N. Kang, and T. R. Sheltami, "EAACK—A Secure Intrusion-Detection System for MANETs," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 3, pp. 1089–1098, 2013.

[12] N. Marchang, R. Datta, and S. K. Das, "A Novel Approach for Efficient Usage of Intrusion Detection System in Mobile Ad Hoc Networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 2, pp. 1684–1695, 2017.

[13] T. Nguyen and G. Armitage, "A survey of techniques for internet traffic classification using machine learning," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 4, pp. 56–76, 2008.

[14] M. Mahmoud and X. Shen, "An Integrated Stimulation and Punishment Mechanism for Thwarting Packet Dropping Attack in Multihop Wireless Networks," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 8, pp. 3947–3962, 2011.

[15] K. Kosek-Szott, M. Natkaniec, and A. R. Pach, "A new busy signal-based MAC protocol supporting QoS for ad-hoc networks with hidden nodes," *Wireless Networks*, vol. 19, no. 6, pp. 1135–1153, 2013.