Interference exposure System for unauthorized User in Wide Network

[a] Pradosh Chandra Patnaik, [b] M V Ramana Murthy, [c] S B Kishor
[a]Research Scholar in Dept. Of Computer Science, Gondwana University, Godchiroli, Maharashtra, India
[b] Head, Dept of Computer Science, Osmania University, Hyderabad, Maharashtra, India
[c] Head, Dept. Of Computer Science, Sardar Patel Mahavidyalaya, Chandrapur, Maharashtra, India

Abstract

Wide network play a crucial role in the lives of people at work, home, and public places. Because of the widespread implementation of wide network, providers deploy Wide Local Area Network (WLANs) to provide broadband access to the Internet. The rapid growth rate of public wide LANs is making the Internet available to people at areas where people tend to congregate. Users within range of a public wide LAN hotspot, such as an airport or hotel, can access e-mail and browse the Internet either free or mostly for a fee when it is not offered freely. However, as the number of users soars, so does the risk of possible unauthorized. Unauthorized user can substantially increase his share of the bandwidth, at the expense of other paid users, by slightly modifying the driver of the network adapter. As the uses of such networks grow, users will demand secure yet efficient, low-latency communications. Interference exposure is one of the key techniques behind protecting a network against intruders. Many Interference Exposure Systems (IESs) have been designed for wired networks. Most of these IES systems did not produce the expected results when applied to wide networks.

KEYWORDS: IES, Unauthorized, Selfish behaviour, Cheater, Wide LAN

I. INTRODUCTION

THE widespread deployment of the IEEE 802.11 hotspot will continue increasing as long as the revenue from using hotspots will increase. Therefore commercial use of these networks has already revealed a set of problems related to security [10]. One major challenge neglected so far by the research community is Media Access Control (MAC) Layer selfish behavior, an area that only recently captured the attention of the research community. The focusing was frequently done in the malicious behaviour from most previously researched papers in the WLAN field [16]. This only captured one side of the problem: it is like describing the stick without mentioning the carrot.

IEEE 802.11 standard for Wide networks have inherent vulnerabilities that are not easily preventable. Also the MAC layer protocol in the Wide LAN has its vulnerabilities. The deviation from legitimate protocol operation in wide networks as mentioned before has received considerable attention in recent years. The distributed nature of wide networks with devices that are gradually becoming essential components in our everyday life justifies the rising interest on that issue. Notice that the architectural organization of wide networks in distributed segregated user communities raises issues of compliance

with protocol rules. In addition, users are clustered in communities that are defined on the basis of proximity, common service or some other common interest. Since such communities are bound to operate without a central supervising entity, no notion of trust can be presupposed. Furthermore, the increased level of sophistication in the design of protocol components, together with the requirement for flexible and readily reconfigurable protocols has led to the extreme where wide network adapters and devices have become easily programmable. As a result, it is feasible for a network peer (station) to tamper with software and firmware, modify its wide interface and network parameters and ultimately abuse the protocol. This situation is referred to as protocol unauthorized. The goals of misbehaving station range from exploitation of available network resources for its own benefit up to network disruption.

A misbehaving station in the wide network can deliberately misuse the MAC protocol to gain bandwidth at the expense of other stations. This unauthorized as mentioned can be done from changing the network driver either by manipulating some of the standard parameters or by departing from the communication procedures. As a result it will lead to many benefits (for the misbehaving station) like the following [18]:

- It can result in significant bandwidth gains as it directly deals with the wide medium. Therefore, it is more efficient than unauthorized at the network and transport layers.

- It is hidden and independent from the upper layers and hence cannot be detected by any mechanism designed for those layers. However, it can be combined with upper layer unauthorized to increase the impact.

- It is always usable, since all wide stations use the same IEEE 802.11 MAC protocol.

The solution to the problem is to provide a reliable exposure of such unauthorized instances, which would eventually lead to network defence and response mechanisms and isolation of the misbehaving station.

This paper presents a designed System Alarm IES Tool for the selfish behaviour in IEEE 802.11 MAC layer of WLAN [11, 7]. It will provide a general bound for the worst-case attack scenarios in wide networks for the case of the existence of one or many intelligent adversaries. We will also previewing the approaches to modelling and extend the studies of such behaviours for the exposure system.

The organization of this paper is the following. The next Section discusses existing work in the areas of the IEEE 802.11 MAC unauthorized and the exposure of such attacks. Section 3 presents an overview of (i) the IEEE 802.11 wide networks MAC layer and (ii) the TCP congestion control at the transport layer which is an essential for better understanding of the unauthorized problems. In Section 4 we will explore this space of

MAC Layer selfish behaviour, specifying unauthorized techniques that can be used by the cheaters. In Section 5 we presents in detail the counter technique can be used as a solution for selfish behaviour in a way that is transparent to the operation of the network compared to some proposed techniques that need modifications to the existing standard. Section 6 will focus in describing the implementation of the System Alarm IES Tool used for detecting MAC cheating. Conclusions will be in Section 7.

II. RELATED WORK

Although the protocol unauthorized has been studied in various scenarios in most communication layers and under several mathematical frameworks, the problem of Media Access Control (MAC) layer unauthorized is relatively new and unexplored in the literature. Various solutions have been proposed to routing layer unauthorized in the ad hoc networks type of Basic Service Set (BSS) that belongs to the wide Local Area Networks (WLANs). However, the problem we consider in this paper is focusing on finding the proper solution to the MAC layer which is too different to make those proposed solutions eligible here. In order to have a brief idea about some of those proposed solutions, this section is introduced.

i) Kyasanur and Vaidya [20] present their idea as a proposed solution to the MAC Layer unauthorized. They address the MAC layer unauthorized using exposure and correction mechanisms. Their main idea is to let the receiver assign and send back-off values to the sender in Clear to Send (CTS) and Acknowledge (ACK) frames and then use them to detect potential unauthorized. The latter is handled using a correction scheme that adds to the next back-off a penalty that is a function of the observed unauthorized. This solution is efficient for the unauthorized of MAC Layer, but at the expense of the following issues:

- First it requires a modification of the IEEE 802.11 MAC protocol in a way that is incompatible with the current standard. Such an approach is practically unfeasible and also brings to the surface the problem of compatibility of the current standard.

- It gives control to the receiver over the sender, by making the former (receiver) assign back-off values to the latter (sender) in both the exposure and the correction schemes. Hence the proposed approach opens the door to new unauthorized techniques in the MAC Layer, including the situation of misbehaving receivers and the collusion that may occur between the sender and receiver.

- It creates communication and computation overhead if applied in the Wide Local Area Networks (WLANs). The first is due to the addition of new frame header fields and the second to the exposure and correction schemes that have to compute back-off values and, in some cases, penalties for each individual frame of the sending station (in the infrastructure case, all this load will be centralized at

the AP, where applying it to the Ad hoc networks will be a different story).

- It considers only stations with backlogged UDP traffic to detect unauthorized. But if the misbehaving station generates traffic with an interframe delay, the latter may result in the measured back-off being larger than the assigned one and hence leave the cheater undetected.

ii) Konorski [19] considers an ad hoc network in which all stations hear each other and he proposes a unauthorized-resilient back-off algorithm based on game theory. As it requires a new back-off mechanism, different from the current standard, this solution is not practical for current hotspots. It is worth noting that none of the previous two works include a real implementation of the proposed algorithms.

iii) The authors in [18] focus in MAC layer unauthorized in wide hotspot communities. They propose a sequence of conditions on available observations for testing the extent to which protocol parameters have been manipulated. The advantage of the scheme is its simplicity and easiness of implementation. This paper was an important source of inspiration for this paper.

Interference exposure systems [6] are also relevant to the MAC layer, although they handle security flaws rather than protocol unauthorized. A commercial example of these systems is AirDefense Guard, in which distributed sensors placed near APs monitor the wide medium and send reports to a central server.

III. BACKGROUND

This section presents a survey of a few important background topics, including: (a) how the MAC Layer works according to the IEEE 802.11 Standard, (b) how the TCP protocol at the transport layer behaves to solve the problem of traffic congestion, and (c) the difference between the ACK of the MAC layer compared to the ACK of the TCP protocol which belongs to the transport layer. We direct our focus to the above subjects to get better understanding of the MAC unauthorized problems in our paper.

3.1 IEEE standard MAC Layer

IEEE 802.11 is the most prevalent standard for wide LANs [8]. This standard specifies a common *medium access control* (MAC) and several physical layers for wide LANs. The MAC Layer is specified in the 802.11 standard as the brain of the wide network. It takes the responsibility of controlling the access to the shared air medium by directing the 802.11 physical layer to perform the tasks of sensing the medium, transmission, and receiving of 802.11 frames. However, the MAC layer uses a coordination function to control the access to the medium. This coordination function can be a DCF or PCF, both of which are defined below (see Figure 3.1).

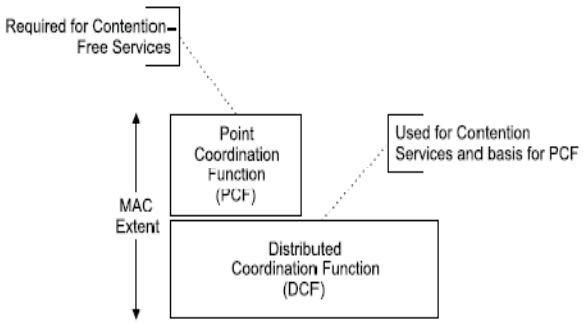

Figure 3.1: MAC Architecture [8].

3.2 TCP Congestion Control

When the load of the network traffic for any given network starts to become greater than the network ability to handle, the case of congestion begins to appear [17]. Although the third layer of the OSI Model (network layer) partially uses its function to manage the congestion, most of the remaining work is done by TCP because the practical solution to the situation of congestion is to slow down the data rate. In theory, there is a principle borrowed from physics that can be used as a solution to the congestion situation. This principle is called *the law of conservation of packets*. The idea is to delay the injection of new packets into the network until an old one leaves the network (i.e., the packet is delivered to its destination). TCP tries to achieve the goal of reducing the congestion by dynamically manipulating the window size. Detecting the congestion is the first step in managing its appearance. In the past, the congestion situation was too hard for network administrators to discover. Every time a packet is lost there is a timeout and this loses of packet could have been caused by either (1) noise on a transmission line or (2) packet discard at a congested router. Knowing the difference was difficult. Nowadays, packet loss due to errors in the transmission line is relatively rare because most long-haul trunks are fiber (although wide networks are a different story). Consequently, most transmission timeouts on the Internet are due to congestion. All the Internet TCP algorithms monitor timeouts for signs of trouble the way miners watch their canaries due to the assumption that timeouts are caused by congestion.

Describing first what TCP does is essential, before discussing how it reacts to congestion to try to prevent congestion from occurring in the first place. A suitable window size has to be chosen, when a connection is established. Based on its buffer size, the receiver can specify a window. Due to buffer overflow at the receiving end, problems will not occur if the sender sticks to this window size, but due to internal congestion within the network they may still occur. This problem is illustrated hydraulically In Figure 3.6. Figure 3.6(a) shows a small-capacity receiver waiting at the end of a thick pipe. No water will be lost as long as the sender does not send more water than the bucket can contain.

Figure 3.6(b) shows certainty that the bucket capacity is not the limiting factor, but the

internal carrying capacity of the network is the one. If too much water comes in too fast, it will back up and some will be lost (in this case by overflowing the tunnel).

The network capacity and receiver capacity are the two potential problems which should be realized as the solution to the Internet and be dealt with separately. In order to do that, two windows must be maintained by each sender: first is the window granted by the receiver and a second one is the *congestion window*. The number of bytes the sender may transmit is reflected by each of the two windows. The number of bytes that may be sent is determined by the minimum of the two windows. Thus, the effective window is the minimum of what the receiver thinks is all right and what the sender thinks is all right. Therefore, if the receiver says "Send 8 KB" but the sender knows that bursts of more than 4 KB clog the network, the sender sends 4 KB. But, if the receiver says "Send 8 KB" and the sender knows that bursts of up to 32 KB get through effortlessly, it sends what the receiver asks for, i.e., the full 8 KB as requested.

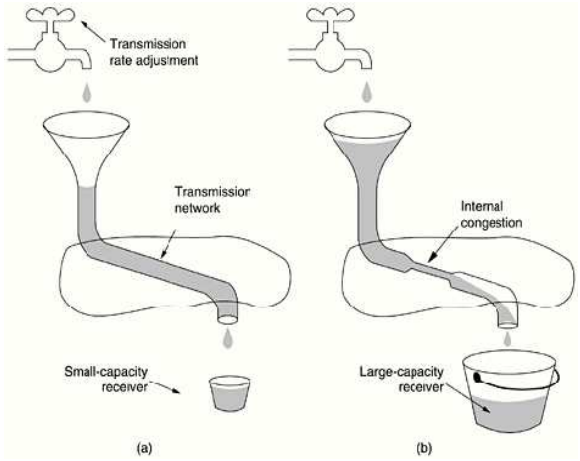

Figure 3.2: (a) A fast network feeding a low-capacity receiver. (b) A slow network feeding a high-capacity receiver [17].

The size of the congestion window is initialized to the size of the maximum segment in use on the connection by the sender, after establishing the connection. One maximum segment will be then sent through the connection. Before the timer goes off if this segment is acknowledged, it sends two segments after it adds one segment's worth of bytes to the congestion window to make it two maximum size segments. The congestion window is increased by one maximum segment size as each of these transmitted segments is acknowledged. When the congestion window is *n* segments, the congestion window is increased by the byte count corresponding to *n* segments, if all *n* are acknowledged on time. Therefore, each burst acknowledged doubles the congestion window. When neither a timeout occurs nor the receiver's window is reached the congestion window will keep growing exponentially. The idea is that if bursts of size, say, 1024, 2048, and 4096 bytes work fine but a burst of 8192 bytes gives a timeout, the congestion window should be set to 4096 to avoid congestion. No bursts longer than 4096 will be sent, as long as the congestion window remains at 4096, no matter how

much window space the receiver grants. This algorithm is called *slow start*, but it is not slow at all (as it is growing exponentially). All TCP implementations are required to support it.

Now considering the Internet congestion control algorithm, in addition to the receiver and congestion windows, it uses a third parameter called the *threshold* (initially 64 KB). The threshold is set to half of the current congestion window when a timeout occurs, and the congestion window is reset to one maximum segment. Except that exponential growth stops when the threshold is hit, slow start is used to determine what the network can handle. Therefore, successful transmissions grow the congestion window linearly (one maximum segment grow for each burst) instead of one per segment. Thus, this algorithm is guessing that it is probably acceptable to cut the congestion window in half, and then it gradually works its way up from there.

Figure 3.8 shows an illustration of how the congestion algorithm works. The maximum segment size here is 1024 bytes. Initially, the congestion window was 64 KB, but the threshold is set to 32 KB and the congestion window to 1 KB for transmission 0 here because a timeout occurred. Until it hits the threshold (32 KB) the congestion window will continue to grow exponentially. Starting then, the congestion window grows linearly. As illustrated in Figure 3.8, transmission 13 is unlucky as a timeout occurs. The threshold is set to half the current window (the half is 20 KB after it was 40 KB), and slow start is initiated all over again. When the acknowledgements from transmission 14 start coming in, the first four each double the congestion window, but after that, growth becomes linear again. The congestion window will continue to grow up to the size of the receiver's window, if no more timeouts occur. At that point, as long as there are no more timeouts and the receiver's window does not change size it will stop growing and remain constant.

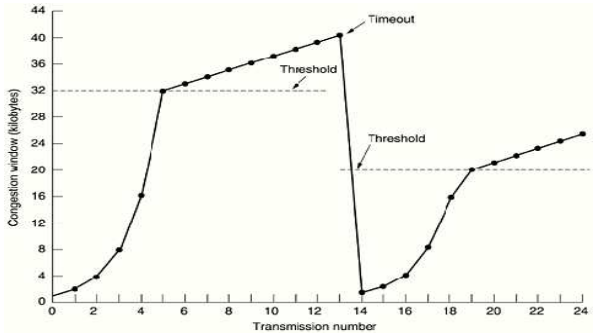

Figure 3.3: An example of the Internet congestion algorithm [17].

IV. POSSIBLE TECHNIQUES USED BY CHEATER

In this section the focus will be on possible techniques used for MAC greedy behaviour, which happens when the cheater changes the operation of the IEEE 802.11 protocol either by not obeying to the communication procedures or by changing parameters defined in the standard [18].

Several studies have shown that most of the traffic (around 90%) flowing over public wide LANs is TCP and is mainly downlink. Therefore, according to the type of traffic they target either TCP or UDP, it is important to distinguish unauthorized techniques. In the following we describe greedy attacks on uplink traffic (both TCP and UDP) and downlink TCP traffic. Cheater attacks against downlink UDP traffic are much more difficult to perpetrate and will not be considered in this paper.

4.1 Uplink traffic

- A greedy station can selectively scramble frames sent by other stations in order to increase their contention windows. The frames to be targeted can be the following:

1. CTS frames. In this case the cheater hears an RTS frame sent by another station to the Access Point (or to any other station in case of Add-hoc network) and intentionally causes collision and loss of the corresponding CTS frame in order to prevent the frame exchange sequence. As a result, the channel becomes idle after the corrupted CTS, the station whose CTS was jammed doubles its contention window, and the cheater gets a higher chance to send his data.

2. ACK and DATA frames. Although jamming these frames does not result in saving the data frame transmission time, it causes the contention window of the ACK destination (i.e., the DATA source) station to be doubled and consequently makes the latter select larger backoffs. As in the previous case, the cheater increases his chances to get access to the channel.

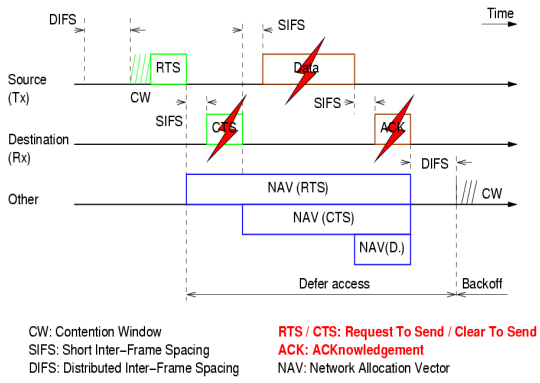

Figure 4.4 Frame Scrambling

- A greedy station can manipulate protocol parameters to increase bandwidth share:

1. When the channel is idle, transmit after SIFS but before DIFS.

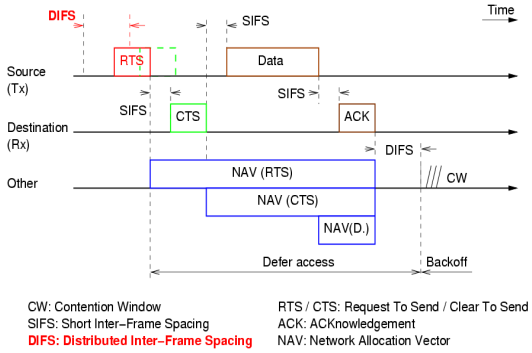

Figure 4.4 Short DIFS

2. When sending RTS or DATA frames, set the duration field (in the frame headers) to a high value; in this way, as the stations in range set their NAVs with that value, they will refrain from contending to the channel during all that time.

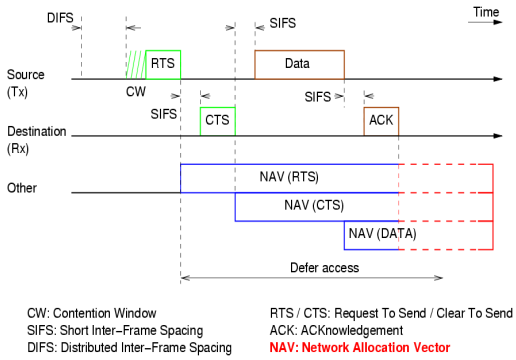

Figure 4.4 Oversized NAV

3. Reduce the back-off time. This can be done by choosing a small fixed contention window; thus, the back-off is always chosen from this small window.

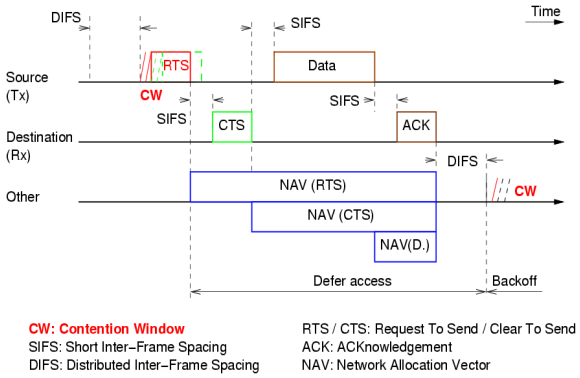

Figure 4.4 Backoff Manipulation

A cheater may also combine several of the above techniques or adaptively change hit unauthorized to avoid being detected.

4.2 Downlink traffic

- In the case of the downlink traffic, the cheater will attempt to increase the share of traffic sent to him through the AP (or through any add-hoc station that pretends as AP), thus increasing the number of packets destined to it in the AP's queue; to achieve this goal, he will target the protocols responsible for filling this queue. We can distinguish two types of sources (e.g., Web servers) sending traffic to wide stations through the AP:

  1. UDP source: since UDP requires no acknowledgements from the receiver and hence cannot be affected by channel conditions, attacking UDP traffic is pointless.

  2. TCP source: on the contrary to the above case, the TCP traffic rate reacts to the channel conditions by using congestion windows and acknowledgements from the receiver. Hence an attack can be mounted on the TCP traffic by exploiting the congestion avoidance mechanism and reducing the source rate until eventually shutting down the flow.

Downlink attacks are relatively less intuitive and require more "effort" from the cheater's side to increase his share of the bandwidth, and from the AP's side to detect the unauthorized. Deep looking on the closed-loop nature of TCP flows, we found their impact goes beyond the local area (the hotspot and associated nodes) to reach remote servers. Consider the topology in Figure 4.1 and the typical following scenario: Two mobile nodes Bc and Ac are connected to the Internet via the AP. Ac and Bc download large files from two remote servers, As and Bs, respectively. Both downloads uses FTP/TCP. To increase his download data rate, the cheater (Ac) can use the following two techniques to reduce Bs's data rate, thus freeing more bandwidth for himself at the AP (or at any common bottleneck between the servers and the AP):

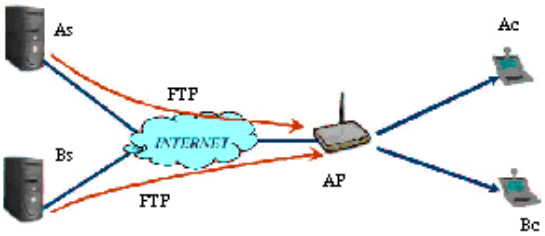

Figure 4.1 stations Ac (cheater) and Bc downloading from servers As and Bs [18].

- Ac jams the TCP-ACKs from Bc to the AP, so they never reach the server Bs. As TCP-ACKs get lost (jammed), Bs decreases its sending data rate, using TCP congestion control, and ends up killing the connection. At the AP, Bc's share of the bandwidth decreases, leading to an increase of the data rate from As to Ac.

In the previous technique, the AP can still hear the collisions/jamming and may end up

detecting Ac based on the number of retransmissions of Bc. Another option for Ac consists in jamming AP's frames destined to Bc, therefore reducing Bs's data rate, without being heard by the AP. However, Ac's packets share the same queue as Bc's packets at the AP. While jammed frames get repeatedly retransmitted by the AP, Ac's packets get delayed in the queue, and his data rate (from As) decreases as well. To prevent AP's retransmissions and the queuing delays, Ac sends forged MAC-ACKs on behalf of Bc for the jammed packets. This avoIES retransmissions at the AP, while still reducing the data rate from Bs. Furthermore, Ac can jam only part of the AP's frames to Bc, saving his battery power and making exposure even harder.

V. COMPONENTS OF SYSTEM ALARM IES TOOL

Several approaches can be envisioned to counter unauthorized techniques [7, 18 and 11] which are presented in section 4. The proposed System Alarm IES Tool is one of suggested solutions. Given the number of possible attacks and their independence, System Alarm has a modular architecture, depicted in Figure 5.1.

System Alarm needs to be implemented only on AP in case of infrastructure network as a network based IES or in any one of the communicating stations in case of the Add-hoc network as a host based IES. The System Alarm periodically collects traffic traces of the active user stations during short intervals of time called monitoring periods

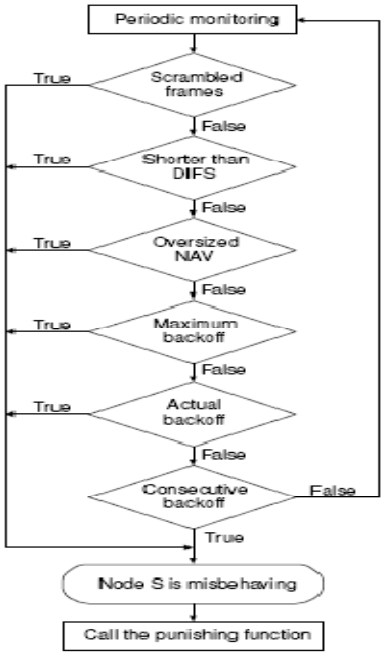

Figure 5.1: The Modular Architecture of System Alarm.

Exposure of the intended collisions

The goal of this test is to detect the unauthorized techniques that rely on frame

scrambling; they correspond to the first attacks described in the uplink and downlink parts of section 4.

5.1 Scrambled frames

In order to gain a significant share of the common wide bandwidth using CTS/ACK/DATA scrambling, the cheater has to scramble a relatively large percentage of CTS, ACK, or DATA frames sent by other stations. As a result, its average number of retransmissions will be less than that of other stations, and it can be detected using Test 1 (for this test as for the following ones, if the inequality holds, it means that a greedy attack is probably taking place. see Table 5.1).

Table 5.1: Test 1.

Test 1 Scrambled frames
$num\_rtx(Mi) < \Phi \times E_{j \neq i}[num\_rtx(Mj)]$

In this test, num_rtx(*M*) is the number of times station *M* retransmitted its last frame successfully received by the AP. Φ is a tolerance parameter with a value between 0 and 1; it is applied to the average number of retransmissions of all "other" stations, $E_{j \neq i}$.

The developed System Alarm IES tool can detect a retransmission by observing a repeated sequence number in the header of RTS or DATA frames when the corresponding CTS or ACK frames are scrambled, respectively. In the case of DATA frames, one might argue that the AP would not be able to distinguish retransmissions because the DATA frames are scrambled. However, the cheater cannot scramble the headers of these frames; otherwise it cannot know whether a given frame is destined to himself.

A potential cause of false positives (leading to a system alarm detecting a unauthorized station when there is no one misbehaving) for this test could be the bad channel conditions that lead to frame loss and retransmission. To avoid this pitfall, the Φ tolerance parameter must be chosen carefully.

Exposure of manipulated protocol parameters

The following paragraphs will address the unauthorized techniques that alter protocol parameters.

5.2 "Shorter than DIFS"

The AP can monitor the idle period after the last ACK and distinguish any station that transmits before the required DIFS period. After having observed this unauthorized repeatedly for several frames from the same station, the AP can make a reliable decision according to Test 2 (see Table 5.2).

Table 5.2: Test 2.

Test 2 Shorter than DIFS
Idle_time_after_ACK(Mi) < DIFS

5.5 "Oversized NAV"

By measuring the actual duration of a transmission (including the DATA, ACK, and optional RTS/CTS) and comparing it with the duration field value in the RTS or DATA frame headers, the AP can detect a station that regularly sets the duration field (and therefore the NAV of listening stations) to very large values. In Test 3 (see Table 5.3), the tolerance parameter A (greater than 1) ensures that the AP does not mistakenly incriminate well-behaved stations.

Table 5.3: Test 3.

Test 3 Oversized NAV
$A \times actual\_duration(Mi) < duration(Mi)$

Back-off manipulation

In the following we will address the unauthorized techniques that belong to the back-off manipulation since they are the easiest to implement and the hardest to detect.

5.4 "Maximum back-off"

Since the IEEE 802.11 protocol selects back-offs randomly from the range [0, *CW* - 1] (where CW depends on the number of retransmissions), the maximum selected back-off $max_{bkf}(Si)$ over a set of frames sent by a given station should be greater than or equal to CW_{min} -1, if the number of samples is large enough. Test 4 uses this property to suspect stations whose maximum back-off over a set of samples is smaller than a threshold value $threshold_{maxbkf}$ (see Table 5.4).

Clearly, a tradeoff exists between the number of samples and the threshold; if the IES system increase the threshold (its largest value is CW_{min}), it have to increase the number of sampled back-off's to get more distinct values and thus avoid false positives. In the designed System Alarm IES Tool, a threshold equal to $CW_{min/2}$ is used; thus, the test works if the reduced contention window is in [0, $CW_{min/2}$ -1].

Table 5.4: Test 4.

Test 4 Maximum back-off
$max_{bkf}(Si) < threshold_{maxbkf}$

Unfortunately, this check may be easily tricked by a clever cheater that succeeds at making the monitor observe in every sample at least one back-off value larger than or equal to the threshold; channel conditions can also yield a similar result and thus make

the check fail. Thus, the maximum back-off check is only auxiliary to the following two tests.

5.5 "Actual back-off"

Test 5 consists in measuring the actual back-off, as shown in Table 5.5. The main procedures of the test can be summarized as follows:

- If between two transmissions from a station *M* there are no collisions, we assume that *M* spent all its idle time backing off (although it may be just part of the *M*'s inter frame delay, if it is transmitting at low data rates). Then estimating this back-off can be done by computing the sum as illustrated (see Figure 5.2).

- If a collision happens, it may be more difficult to know the identities of the senders of the colliding frames and hence which are the stations whose measured *actual back-off* should be updated. To avoid complexity, collisions are simply not taken into account; in case of collisions, neither the current back-off nor the next one is measured for any station.

Table 5.5: Test 5.

Test 5 Actual back-off
$B_{ac}\ [Mi\ ] < \alpha_{ac} \times B_{acnom}$

In Test 5 (see Table 5.5), $B_{ac}[Mi]$ denotes the average actual back-offs (observed by the AP) of station *Mi*. B_{acnom} is the nominal back-off value, which is equal to the average back-off of the AP, assuming it has enough traffic to compute this value. The α_{ac} ($0 < \alpha_{ac} \leq 1$) parameter is configurable according to the desired true positive (correct exposure) and false positive (wrong exposure) percentages (for example, a value of $\alpha_{ac} = 90\%$ may be used in our proposed tool).

As it collects no data during collisions, the actual back-off test measures back-off's that are selected only from the $[0, CW_{min} -1]$ range. Due to its mechanism, this test fails to detect a unauthorized case if the cheater has inter frame delays (e.g., a TCP source using congestion control). In fact, the test measures these delays instead of back-off's because it adds up the idle periods between transmissions from the same source. The solution to this problem is provided by the consecutive back-off test.

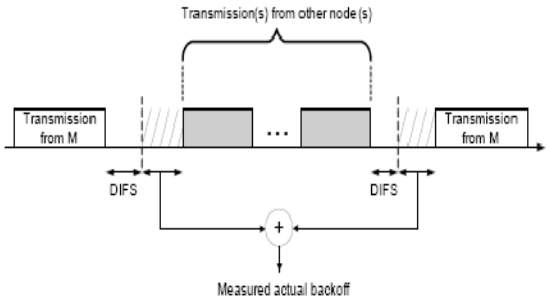

Figure 5.2: Measurement of the actual back-off. Transmissions from M are interleaved with one or more transmissions from other nodes (including the AP). The transmission includes, in addition to the DATA frame, all the control frames such as RTS, CTS, and ACK, as well as the interleaving idle periods of SIFS and DIFS. The measured value is the sum of all idle intervals (not including inter frame spaces) between two transmissions from M [7].

6.1 The wanted data types

Before starting the data collection stage the need to describe the types of these data is important, otherwise the developer of the system will face an ambiguity condition due to the huge and deferent types of available collected data. This results in degrading the system performance plus the difficulty to know from where to start the analyzing stage? So discarding the unwanted data is important to assure good results.

This section examines in detail only three of the presented techniques in system implementation is more than enough to reach the purpose of this paper. From the previous described techniques the selected three in the implementation all the way until the end of this section will be as follows:

- DIFS unauthorized
- Max back-off unauthorized
- Scrambled Frames unauthorized

Therefore beginning the study of the wanted data types of the above three chosen techniques is required.

All the different types of frames DATA, ACK, RTS and CTS are divided into frame header plus the encapsulated data to be delivered (just for DATA frame only) and the Frame Check Sequence (FCS). The frame header have variegated information's existing in many fields with difference in their size measured by octets 8 bit each (see Figure 6.1).

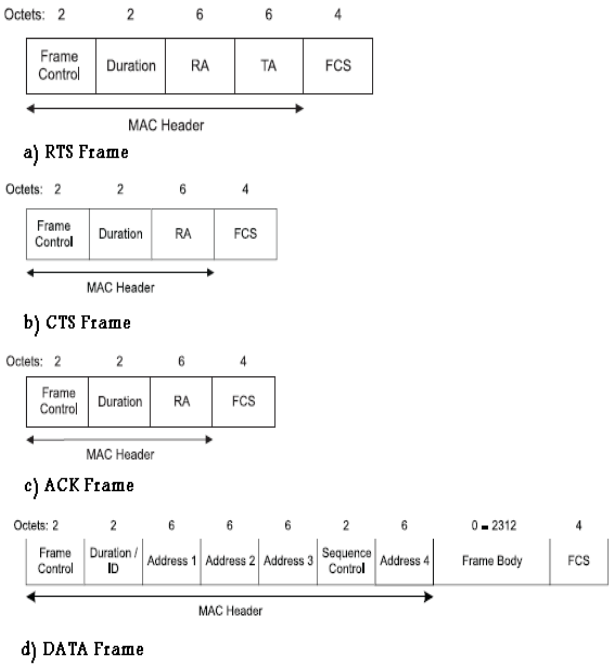

Figure 6.1: Types of Frame Format [8].

6.1.1 The Wanted fields for DIFS Unauthorized

Test 1 is used to discover the unauthorized of shorter than DIFS where the Table 5.2 in section 5 shows the two parts of equation.

Idle_time_after_ACK(Mi) < DIFS

The first part is idle time after the acknowledgement of previous transmission of any transmitting station will be derived from multiple fields found in frame header of involved transmitted frames. The second part DIFS is the defined standard idle time used in transmissions. The address field of the transmitting station is needed to distinguish the transmitting stations from each other in order to know the misbehaving one. The second wanted field will be the field used to know the type of the frame. This can be derived from the *Frame Control* field which is used to identify the function of the frame. There are three types of frames: control, data, and management. The management frame is used mainly for requesting join/disjoin to or from the WLAN network, where the IES system will be used for detecting the misbehaving station from the stations that already joined the network. Thus the management frame type will be neglected in the system implementation. The control frames will be the ACK, RTS and CTS where the data frame type is the DATA. All these types of frames (DATA, ACK, RTS and CTS) need to be known for the IES system. The received time of the frame is required for calculating the idle time spent before starting the new transmission. Also the Bit rate is needed in the calculation for the transmission time of the determined frames. The Bit rate can be known from the preamble frames added at the physical layer. In addition the recording of the transmission order in the time scale is definitely important to know which transmission comes first and which one comes later.

6.1.2 The Wanted fields for the Max Back-off

Test 2 previewed the invented way to detect this type of unauthorized, where the parts of the equation are:

$max_{bkf}(Si) < threshold_{maxbkf}$

All the above mentioned types of fields and information's in DIFS unauthorized are also required in order to detect the unauthorized of the Max Back-off. The threshold value is the defined parameter from the network administrator and it should be chosen carefully to avoid false positive alarms.

6.1.3 The Wanted fields for Scrambled frames

Test 3 is for detecting the different scrambled techniques used for unauthorized is as shown:

$num\_rtx(Mi) < \Phi \times E_{j \neq i}[num\_rtx(Mj)]$

The *Mac Address* field is required for distinguishing transmitting stations also incrementing the counter of the retransmission required. Every time a retransmission happens this can be discovered from the sub control *Retry* field from the *Frame Control* field.

6.2 Exposure of Unauthorized

When the appropriate thresholds are selected for the exposure of the unauthorized station which was described in section 5, just run the exposure algorithm by clicking the analyze button (see Figure 6.8). Then the algorithm starts by opening the generated traffic database and making the check for every station with the approved tests (the agreed tests for the simulation are the DIFS, Maximum back-off, CTS/DATA/ACK scrambling unauthorized tests) and doing the required calculations for time and the other factors according to the previous equations in section 5 used to discover the misbehaving one.

VI. CONCLUSION

In summary, to build a highly secure wide network, we need to deploy Interference exposure techniques (the second wall of defence after the Interference prevention measures like firewall, encryption and authentication mechanism [10]) to this new environment, from their original applications in the fixed wired network. An IES is not a replacement for a firewall, it is just a layer of the total security onion. Although some firewalls have Interference exposure capabilities, they are typically able to detect fewer attacks than full-fledged IES. This paper presented a tool for Interference exposure in mobile network as proposed solution to catch the unauthorized user that benefit from the weakness in the MAC layer protocol. The key feature of the System Alarm IES tool its full compliance with existing standards, and its ability to identify the cheaters.

REFERENCES

[1] R. Lippmann et. al., "Evaluating Interference Exposure Systems: The 1998 DARPA Off-Line Interference Exposure Evaluation", Proceedings of DARPA Information Survivability Conference & Exposition II, Volume: 2, 1999, pp. 12-26.

[2] J. Haines, L. Rossey, R. Lippmann, R. Cunningham, "Extending the DARPA Off-Line Interference Exposure Evaluations", Proceedings of DARPA Information Survivability Conference & Exposition II, Volume: 1, 2001, pp. 35-45.
[3] A. Siraj, S. Bridges, R. Vaughn, "Fuzzy Interference Exposure", Joint 9th IFSA World Congress and 20th NAFIPS International Conference, Volume: 4, 2001, pp. 2165-2170.
[4] M.C. Bernardes and E. Santos Moreira, "Implementation of an Interference Exposure System based on Mobile Agents", Proceedings of International Symposium on Software Engineering for Parallel and Distributed Systems, 2000, pp. 158-164.
[5] G. Helmer, J. Wong, V. Honavar, L, Miller, "Lightweight Agents for Interference Exposure", Technical Report, Dept. [of Computer Science, Iowa State University, 2000.
[6] Y. Zhang and W. Lee, "Interference Exposure in Wide Ad-Hoc Networks", Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, MobiCom'2000, pp. 275- 83.
[7] M. Raya, J.-P. Hubaux, and I. Aad. "DOMINO: A System to Detect Greedy Behavior in IEEE 802.11 hotspots", In Proceedings of the ACM Conference on Mobile Systems, Applications and Services (MobiSys), Boston, Massachusetts, USA, June 2004.
[8] IEEE Standard for Wide LAN-Medium Access Control and Physical Layer Specification, P802.11, 1999.
[9] A. Balachandran, G. Voelker, P. Bahl, and P. Rangan. Characterizing user behavior and network performance in a public wide LAN. In *Proceedings of ACM SIGMETRICS'02*. ACM Press, June 2002.
[10] W. Stallings, "Cryptography and Network Security Principles AND Practices", 3rd Ed, Prentice Hall PTR, 2003.
[11] A. Cardenas, S. Radosavac and J. Baras, "Exposure and Prevention of MAC Layer Unauthorized in Ad-Hoc Networks", *In Proceedings of ACM SASN'04*, ACM Press, October 2004.
[12] A. Santamaria and F. Hernandez, "Wide LAN Standards and Applications", Artech House, 2001.
[13] R. Prasad and L. Munoz, "WLAN and WPANs towards 4G wide", Artech House, 2003.
[14] J. Casad, "SAMS Teach Yourself TCP/IP", 2nd Ed, SAMS, 2001.
[15] J. Geier, "Wide Networks first-step", Cisco Press, 2004.
[16] J. Edney and W. Arbaugh, "Real Security: WiFi Protected Access and 802.11i", Addison-Wesley, 2004.
[17] A. Tanenbaum, "Computer Networks", 4rd Ed, Prentice Hall PTR, 2003.
[18] M. Raya, J.-P. Hubaux, I. Aad, and A. Elfawal. "DOMINO: Detecting MAC layer greedy behavior in IEEE 802.11 hotspots", IEEE Transactions on Mobile Computing (TMC), 35(10), 2006.
[19] J. Konorski. "Multiple access in ad hoc wide LANs with noncooperative stations", In *NETWORKING*, volume 2345 of LNCS, Springer, 2002.
[20] P. Kyasanur and N. Vaidya. "Exposure and handling of MAC layer unauthorized in wide networks", In *Dependable Systems and Networks*, June 2003.