Empirical Studies and Queuing Modeling of Denial of Service Attacks against 802.11 WLANs

Chibiao Liu, James Yu and Gregory Brewster

Abstract — The growing popularity of 802.11-based wireless LANs (WLAN) also increases the risk of security attacks. Most studies of WLAN security are on the protection of data integrity, and few studies are addressing the issue of Denial of Service (DoS) attacks. This paper studies two major DoS attacks of authentication request flooding (AuthRF) and association request flooding (AssRF). Our studies show that these DoS attacks cause significant performance degradations and may disconnect the communications. A queuing model is presented to study the attacking mechanisms, and the causes of performance degradations. The analytical results of the queuing model are validated by the simulation model, and both results are consistent with the empirical data. The queuing model analysis leads to the development of four solutions: Request Authentication (RA), Reduction of Duplicate Requests (RDR), Reduction of Response Retransmissions (RRR), and Round Robin Transmission (RTT). We tested these four solutions and collected empirical data to validate the effectiveness of the solutions. A comparison of these four solutions is presented to show their strengths and weaknesses in resolving the attacks.

Index Terms—AuthRF, AssRF, DoS, WLAN, Queuing Model

I. INTRODUCTION

WLANs are widely deployed at home, Small Office/Home Office (SOHO), campus networks, enterprise networks, and hot spots. Meanwhile, security attacks on the WLAN, such as wardriving [1], have also become popular where hackers can easily find and crash wireless access points. A survey from the Computer Security Institute shows that security attacks on WLANs are the only growing threat of computer crimes [2]. In the past, most studies of WLAN security are on the protection of data integrity. However, there is a growing interest in Denial of Service (DoS) attacks against WLANs [3-11]. Our research focuses on four major DoS attacks: Authentication Request Flooding (AuthRF), Association Request Flooding (AssRF), deauthentication flooding, and disassociation flooding. AuthRF floods an access point (AP) with faked authentication requests to consume the AP’s resources and make it deny legitimate access requests and data services. Meanwhile, AssRF floods association requests to the AP and forces it out of service. The 802.11w standard is proposed to resolve DoS attacks due to spoofed deauthentication, or disassociation message [12], but it does not provide solutions to resolve AuthRF and AssRF attacks.

Our research group [4-6] and Ferreri’s research group [10] reported that AuthRF and AssRF DoS attacks could cause serious problems such as performance degradation and denials to new access requests. This paper analyzes TCP/UDP performance degradations via a queuing model, and identifies deficiencies in the 802.11 implementation as specified in the standard. Based on the results of the queuing analysis, we propose and verify four solutions to resolve these deficiencies.

The remaining sections of this paper are organized as follows. Section II presents the experimental design to study AuthRF and AssRF attacks. DoS attack effects on TCP and UDP traffic are presented in Section III. Data and management frame flows under attacks are discussed in Section IV. The development of the WLAN queuing model is presented in Section V. Based on the WLAN queuing model, performance analyses are presented in Section VI. Analytical and simulation results of the WLAN queuing model are discussed in Section VII. Countermeasures to resolve AuthRF and AssRF are presented in Section VIII along with empirical, analytical, and simulation data to validate the solutions. Conclusions are provided at the end of this paper.

II. EXPERIMENTAL METHODOLOGY

An experimental environment to study AuthRF and AssRF attacks is illustrated in Fig. 1. The wireless workstation (WS-1) is associated with the AP. The Ethernet switch connects the AP and the wired workstation (WS-2) together to form a wired LAN. WS-1 and WS-2 are running Windows XP SP2. WS-1 and WS-2 have TCP/UDP data communications via the AP. AuthRF and AssRF target the AP, and they further affect the TCP/UDP traffic going through the AP. The APs studied in the experiment include commercial APs and the open source based HostAP [13]. The TCP/UDP traffic generator (pcattcp [14]) and the traffic analyzer (Wireshark [17]) are installed on WS-1 and WS-2. The hacker runs void11 [15] to launch DoS attacks. The wireless sniffer [16] is used to capture wireless data frames and management frames.
Fig. 1. Experimental settings to study AuthRF and AssRF

In our research, we conducted experiments to study DoS attacks with different combinations of:
1. Eight different APs
2. Two traffic flows: TCP and UDP
3. Two communication directions: Upstream and downstream
4. Two DoS attacks: AuthRF and AssRF

By applying the attacking tool void11, we tested the effects of AuthRF and AssRF on APs from different vendors. Experimental results show that they are all vulnerable to AuthRF and AssRF attacks. Without losing generality, we use Linksys WAP54G with WPA_PSK and 802.11b for the experimental results in this paper.

III. EMPIRICAL STUDY OF AUTHRF AND ASSRF

The first experimental study is to baseline the TCP RTT and UDP packet loss under the normal 802.11b operation with no DoS attacks. We used the maximum packet size of 1460 bytes for TCP and 1472 bytes for UDP. The TCP data rate is 90 frames per seconds (fps), and the UDP data rate is 100 fps without observing packet loss on the receiving side. The RTT measurement is 8.83ms for TCP.

For each attack, the attacking interval is set at 5 seconds. For different attacks, the attacking rates vary from 1.0fps to 10.0fps. We measure the average TCP throughput and RTT during the attacking period. Since TCP throughput and RTT have an inverse relationship, we discuss only the relationship between RTT and the attacking rate to avoid redundancy. The results are shown in Table I and Table II for AuthRF and AssRF, respectively.

Table I and Table II show that AuthRF and AssRF attacks have similar effects on the TCP traffic. RTT increases slowly from attacking rate of 1.0 fps to 5.0 fps. After 7.0 fps, RTT increases dramatically to a few seconds. When the attacking rate increases to 9.0 fps, the TCP traffic stops as RTT is greater than the attacking interval (5.0 seconds)

For the UDP traffic, the sending rate is fixed at 100 fps as discussed earlier. The relationship between UDP packet loss and attacking rates is shown in Table III. The term AuthRFUpPL refers to the upstream UDP packet loss, and the term AuthRFDownPL refers to the downstream UDP packet loss. As we can see from Table III, AuthRF and AssRF have similar effects on the upstream and downstream UDP traffic when attacking rates increase from 1fps to 7fps. When the attacking rate is above 7 fps, the downstream traffic has a high packet loss and stops at 9.0 fps while the high attacking rate (9.0 fps) has little effect on the upstream traffic.

Table IV and Table V show that a low attacking rate has an insignificant effect on TCP/UDP traffic. A high attacking rate has a significant negative effect on TCP and downstream UDP traffic, with the potential to stop the traffic completely. The upstream UDP traffic is resilient to the attacks even under a high attacking rate. Based on Table IV and Table V, we can categorize AuthRF/AssRF attacks as low-load (LL), medium-load (ML), and high-load (HL) DoS attack. These attack types will be further discussed in Section V.

IV. TRAFFIC FLOWS UNDER AUTHRF/ASSRF

Based on the sniffed wireless data, we find that each fake authentication or disassociation request from the hacker’s application layer is retransmitted two times from the hacker’s 802.11 MAC-PHY layer. Thus, the AP receives three copies of the fake request. The data and management frame flow due to the fake request is illustrated in Fig. 2.
Fig. 2 shows that, when processing fake requests, the AP only receives data frames from the wireless or wired user stations (STAs). The AP does not transmit data frames to the wireless user STA. However, the AP is able to transmit response frames to the hacker wireless STA. This observation can be qualitatively explained as follows.

Under the AuthRF/AssRF attack, the AP receives fake authentication or association requests from the hacker, and it is mandated to send back response frames to this fake wireless STA. As specified in 802.11, the response frame must be acknowledged as well. Since the AP does not receive the 802.11 acknowledgement frame from the hacker, it considers the transmission failed. Then, the AP performs six more retransmissions [18]. Thus, the AP transmits a total of 21 response frames to the hacker wireless STA as illustrated in Fig. 2, and there is no data frame transmitted during this period. Based on the timestamps of sniffed data, we measured the time to process three fake requests and the 21 response frames to be in the range from 100 ms to 150 ms. After processing fake requests and related responses, the AP starts transmitting the queued data frames.

V. QUEUING MODEL OF THE WLAN

The 802.11 standard [18] specifies eight modules for the AP to perform wireless management and data services. Based on the AP service modules, we propose an AP-based queuing model as illustrated in Fig. 3.

The variables involved in the queuing model are grouped in four categorizes:

1. **Dependent variables**: RTT and Packet Loss
2. **Independent variables**: \( \lambda_1, \lambda_2, \lambda_3 \) and D
3. **Measured model parameters**: \( S_1, S_2, S_3, T_0 \) and \( X_2 \)
4. **Calculated model parameters**: \( t_1, t_2, t_3, t_4, t_5, \mu \) and \( Z \)

Above variables are defined in Table VI, which will be referenced in future discussions.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTT</td>
<td>The elapsed time from the transmission of a TCP packet to the reception of a TCP-ACK packet from the receiver</td>
</tr>
<tr>
<td>Packet Loss</td>
<td>The ratio of lost UDP packets at the receiver</td>
</tr>
<tr>
<td>( \lambda_1 )</td>
<td>The arrival rate of data frames to the RX queue</td>
</tr>
<tr>
<td>( \lambda_2 )</td>
<td>The arrival rate of attacking frames to the RX queue</td>
</tr>
<tr>
<td>( \lambda_3 )</td>
<td>The arrival rate of data frames to the TX2 queue</td>
</tr>
<tr>
<td>D</td>
<td>The attack duration that is fixed as 5 seconds in this paper</td>
</tr>
<tr>
<td>( S_1 )</td>
<td>The average service time for the AP_Air RX to process one data frame</td>
</tr>
<tr>
<td>( S_2 )</td>
<td>The average service time for the AP_Air TX to transmit one data frame</td>
</tr>
<tr>
<td>( S_3 )</td>
<td>The normal TCP RTT without DoS attacks</td>
</tr>
<tr>
<td>( X_2 )</td>
<td>The average number of data frames at TX2 during a ML attack or a HL attack</td>
</tr>
<tr>
<td>( t_1 )</td>
<td>The duration for the wireless STA to process a data frame</td>
</tr>
<tr>
<td>( t_2 )</td>
<td>The duration for AP_Air RX to process a data frame</td>
</tr>
<tr>
<td>( t_3 )</td>
<td>The duration for the wired STA to process a data frame</td>
</tr>
<tr>
<td>( t_4 )</td>
<td>The duration for AP_Air TX to process a data frame</td>
</tr>
<tr>
<td>( \mu )</td>
<td>The service rate of AP_Air TX to process data frames</td>
</tr>
<tr>
<td>( Z )</td>
<td>The time point when TX2 does not have enough resources to transmit all queued data frames</td>
</tr>
</tbody>
</table>

Fig. 3 shows that the AP-based queuing model is mainly composed of five elements, which are described as follows:

1. Wireless STA (a legitimate user or a hacker) sends and receives 802.11 frames to and from the AP. These include data frames, management frames, and control frames.
2. The AP_Air RX is responsible for receiving 802.11 frames to the AP. As reported in the literature [19-20], one reception (RX) queue is associated with the AP_Air RX service center. Since the hacker could intentionally flood fake request frames to the AP without following the CSMA/CA protocol, the AP could receive legitimate data frames and fake management frames simultaneously.
3. The AP_Air TX is responsible for transmitting 802.11 frames. The TX1 queue holds 802.11 management frames such as the authentication response frame, association response frame, and deauthentication frame.
The TX2 queue holds 802.11 data frames. As mentioned in the literature [21] and the 802.11 standard [18], management frames have a higher priority than data frames. As a result, data frames in the TX2 queue are processed only when there are no frames in the TX1 queue.

4. The AP_CPU refers to a group of AP’s internal modules. Functionalities of the AP_CPU include processing frames internally, exchanging data with the LAN, and moving frames to TX1 and TX2 queues for transmission. In our experiments, the AP_CPU has a high capacity to process 802.11 frames, and the processing time in the AP_CPU has little variation under the normal and attacking scenarios.

5. The wired STA on the LAN exchanges data with the wireless STA.

We study four wireless communication scenarios in Table VII. It should be noted that because of the TCP-ACK packets, the TCP stream always includes the TX2 queue.

As shown in Table IV and Table V, AuthRF and AssRF have significant effects on TCP traffic and the downstream UDP traffic, but little effect on the upstream UDP traffic. This is because TX2 is involved in TCP traffic and the downstream UDP traffic, where TX2 is congested by AuthRF and AssRF. However, the upstream UDP traffic does not use TX2, so these attacks have little effect on it.

Table IV and Table V show that effects on TCP traffic and the downstream UDP traffic have a strong correlation with the attacking rates. Under different attacking rates, the attacks are categorized as LL, ML and HL DoS attacks as shown in Table VIII. These attack types are defined with variables Tr and Ta, which are defined as follows:

1. **Tr** refers to the time required for transmitting the data frames queued during an attack in the TX2 queue.
2. **Ta** refers to the time available for transmitting queued data frames in the TX2 queue before the next attack.

Tr and Ta are derived as Eq. (1), and (2), respectively [23].

\[
T_r = S_2 \sum_{i=1}^{n} \left( \lambda_3 S_3 \right)^i = S_2 S_3 \frac{\lambda_3}{1 - S_3 \lambda_3} \quad (1)
\]

\[
T_a = \frac{1}{\lambda_2 S_2} \quad (2)
\]

Tr is determined by the average number of data frames in queue after an attack, \(\lambda_3 S_2\), plus additional data frames that enter the queue while these frames are being transmitted. The meaning of Ta is depicted in Fig. 4. Under a low attack rate, Ta > 0, and there is enough time before the next attack to process all queued data frames. However, under a high attack rate, Ta is 0, and there is no time for TX2 to process the queued data frames.

![Fig.4. Illustration of Ta](image)

Based on the values of Tr and Ta, the effect of LL, ML and HL DoS attacks are defined in Table VIII.

### TABLE VII: FOUR TRAFFIC SCENARIOS OF DATA FRAMES

<table>
<thead>
<tr>
<th>Data Stream</th>
<th>Paths through the queuing model</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 UDP Upstream</td>
<td>wireless STA =&gt; AP_Air RX =&gt; AP_CPU =&gt; wired STA</td>
</tr>
<tr>
<td>2 UDP Downstream</td>
<td>wired STA =&gt; AP_CPU =&gt; TX2 =&gt; wireless STA</td>
</tr>
<tr>
<td>3 TCP Upstream</td>
<td>wireless STA =&gt; AP_Air RX =&gt; AP_CPU =&gt; wired STA/(ACK) =&gt; AP_CPU =&gt; TX2 =&gt; wireless STA</td>
</tr>
<tr>
<td>4 TCP Downstream</td>
<td>wired STA =&gt; AP_CPU =&gt; TX2 =&gt; wireless STA/(ACK) =&gt; AP_Air RX =&gt; AP_CPU =&gt; wired STA</td>
</tr>
</tbody>
</table>

### TABLE VIII: AUTHRF/ASSRF ATTACK TYPES VS. TA AND TR

<table>
<thead>
<tr>
<th>Attack type</th>
<th>Ta vs. Tr</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LL DoS attack</td>
<td>0&lt;Tr≤Ta</td>
<td>TX2 queue has enough resources to process queued data frames</td>
</tr>
<tr>
<td>ML DoS attack</td>
<td>Tr=Ta=0</td>
<td>TX2 queue does not have enough resources to process queued data frames</td>
</tr>
<tr>
<td>HL DoS attack</td>
<td>Ta=0</td>
<td>TX2 queue has no resources to process queued data frames</td>
</tr>
</tbody>
</table>

### VI. PERFORMANCE ANALYSES

#### A. Analysis of TCP RTT

Based on Fig. 3 and Table VI, RTT can be expressed as Eq. (3).

\[
RTT = t_1 + t_2 + t_3 + t_4 + t_5 \quad (3)
\]

In general, a high capacity module always has the resource to process incoming requests, and the service time at a high capacity module does not have much variation under different attacking scenarios. \(t_1\) is related to the processing time at the wireless STA, which has a high capacity. The values of \(t_2\) and \(t_3\) are related to the AP_CPU and the LAN, which have high capacities as well. Therefore, the values of \(t_1\), \(t_2\), and \(t_3\) are considered constant in the analytical model, and Eq. (3) is further expressed as Eq. (4).

\[
RTT = t_2 + t_5 + t_0 \quad \text{where } t_0 = t_1 + t_4 \quad (4)
\]

The value of \(t_0\) can be calculated from the normal RTT and data service time at AP_Air RX and AP_Air TX. For the normal TCP communication with a low sending rate (<1.0 Mbps), there is no waiting time on the AP. Thus, \(t_2\) is equal to data service time of \(S_1\) at the AP_Air RX service center, and \(t_3\) is equal to the data service time of \(S_2\) at the AP_Air TX service center. Then, the RTT value (\(T_0\)) of the low traffic TCP data communication is expressed as Eq. (5). From Eq. (5), \(t_0\) is derived as Eq. (6). Combining Eq. (6) and Eq. (4), RTT can be further derived as Eq. (7).
\[ T_d = S_1 + S_3 + t_0 \]  
(5)  
\[ t_v = T_v - S_1 - S_3 \]  
(6)  
RTT = \( t_2 + t_5 + T_0 - S_1 - S_3 \)  
(7)  

The values of \( t_2 \) and \( t_5 \) without DoS attack are derived as Eq. (8) and Eq. (9), respectively; which assumes these act as M/M/1 queues.

\[ t_2 = S_1/(1-\lambda_1 \times S_1) \]  
(8)  
\[ t_5 = S_3/(1-\lambda_2 \times S_3) \]  
(9)  

Under attacks, the value of \( t_2 \) is derived as Eq. (10).

\[ t_2 = S_1/(1-(\lambda_1 + \lambda_2) \times S_1) \]  
(10)  

The values of \( t_2 \) under low load (\( t_{5L} \)), medium load (\( t_{5M} \)), and high load (\( t_{5H} \)) attacks are derived as Eq. (11), Eq. (12), and Eq. (13), respectively. All the derivations for Eq. (8) to Eq. (13) are available at [23].

\[ t_{5L} = S_3 + (\lambda_2/(2\lambda_3)) \times (S_2S_2^2\lambda_2^2 + S_2^2\lambda_3 + S_2 - S_3S_2\lambda_3) \]  
\( + ((1-S_2)S_2^3\lambda_3)/(1-S_3\lambda_3) \) \( + T_0 - S_1 \) \]  
(11)  
\[ t_{5M} = (Z\lambda_3)((\lambda_2/(2\lambda_3)) \times (S_2S_2^2\lambda_2^2 + S_2^2\lambda_3 + S_2 - S_3S_2\lambda_3) \]  
\( + ((1-S_2)S_2^3\lambda_3)/(1-S_3\lambda_3) \) \( + X_2(D-Z)\lambda_2S_2 \) \( + (S_3-1/\lambda_3 \times X_2(X_2 - 1)/2)/(Z\lambda_3 + X_2) + S_1 \) \]  
(12)  
\[ t_{5H} = S_3 + D\lambda_2S_2 \]  
\( + (S_3-1/\lambda_3) \times (X_2 - 1)/2 \) \]  
(13)  

Applying Eq. (7) with \( t_2 \) and \( t_5 \), the average RTT values are derived as Eq. (14), Eq. (15), Eq. (16) and Eq. (17) for the Non-Attack (RTTN), LL attack (RTTL), ML attack (RTTM), and HL attack (RTTH), respectively.

\[ \text{RTTN} = S_1/(1-\lambda_1 \times S_1) + S_3/(1-\lambda_2 \times S_3) + T_v - S_1 - S_3 \]  
(14)  
\[ \text{RTTL} = S_1/(1-\lambda_1 + \lambda_2)S_1 + (\lambda_2/(2\lambda_3)) \times (S_2S_2^2\lambda_2^2 + S_2^2\lambda_3 + S_2 - S_3S_2\lambda_3) \]  
\( + T_v - S_1 \) \]  
(15)  
\[ \text{RTTM} = S_1/(1-\lambda_1 + \lambda_2S_1) + T_v - S_1 \]  
\( + (Z\lambda_3)((\lambda_2/(2\lambda_3)) \times (S_2S_2^2\lambda_2^2 + S_2^2\lambda_3 + S_2 - S_3S_2\lambda_3) \]  
\( + ((1-S_2)S_2^3\lambda_3)/(1-S_3\lambda_3) \) \( + X_2(D-Z)\lambda_2S_2 \) \( + (S_3-1/\lambda_3 \times X_2(X_2 - 1)/2)/(Z\lambda_3 + X_2) \] \( + T_0 - S_1 \) \]  
(16)  
\[ \text{RTTH} = S_1/(1-\lambda_1 + \lambda_2S_1) + D\lambda_2S_2 \]  
\( + (S_3-1/\lambda_3) \times (X_2 - 1)/2 + T_v - S_1 \) \]  
(17)  

B. Analysis of UDP packet loss

For the upstream UDP traffic, we first derive sending and receiving rates, and then calculate the packet loss (PL). The upstream UDP traffic mainly involves the wireless STA, the AP_Air RX, and the wired STA (Fig. 3). Since the wireless STA and the wired STA have high processing capacity, the AP_Air RX is most likely to be the bottleneck of the upstream UDP traffic. In our experiment, the UDP sending rate (\( \lambda_i \)) is fixed, which is equal to the arrival rate to the AP_Air RX queue. The service time at AP_Air RX is \( S_1 \). If \( S_1 \) is less than or equal to the arrival interval of \( 1/\lambda_i \), the UDP departure rate, \( \mu \), is equal to the average arrival rate of \( \lambda_i \). Otherwise, the UDP departure rate is equal to \( 1/S_1 \), see Eq. (18). Because of the high capacity of the LAN module, \( \mu \) will also be the receiving rate at the wired STA. The UDP packet loss is derived as Eq. (19).

\[ \mu = \begin{cases} \lambda_i & S_1 \leq (1/\lambda_i) \\ 1/S_1 & S_1 > (1/\lambda_i) \end{cases} \]  
(18)  
\[ \text{PL}(\%) = \begin{cases} 0 & S_1 \leq (1/\lambda_i) \\ 100((1/S_1 - 1)/\lambda_i) & S_1 > (1/\lambda_i) \end{cases} \]  
(19)  

The downstream UDP traffic mainly involves the wireless STA, the TX2 queue, and the wired STA. Under DoS attacks, data transmission in the TX2 queue is affected by the operations at the TX1 queue, which processes 802.11 management frames. The TX2 queue is the bottleneck for the downstream UDP traffic, and the UDP departure rate (\( \mu \)) is the receiving rate at the wireless STA. \( \mu \) under different DoS attacks is calculated differently.

During the LL DoS attack, the required time \( T_r \) is less than or equal to the available time \( T_a \) (0<\( T_r \leq T_a \)), and the TX2 queue has enough resource to transmit the UDP data frames. Thus, \( \mu \) is equal to the average arrival rate of \( \lambda_3 \). During the HL DoS attack, \( T_a \) is equal to 0, and the TX1 queue is constantly processing 802.11 response frames. Consequently, the TX2 queue has no time to transmit UDP data frames until the DoS attack ends. \( \mu \) is equal to 0 during the HL DoS attack. For the ML DoS attack, the \( T_r \) is greater than \( T_a \) (\( T_r > T_a > 0 \)), and the percentage of UDP frames that could be sent from TX2 is \( TA/Tr \). Thus, \( \mu \) will be \( \lambda_3 \times (TA/Tr) \). In summary, average departure rates under LL, ML and HL attack are expressed as Eq. (20), Eq. (21), and Eq. (22), respectively.

\[ \mu = \lambda_3 \]  
(20)  
\[ \mu = \lambda_3 \times (TA/Tr) \]  
(21)  
\[ \mu = 0 \]  
(22)  

Because the wireless STA is a high capacity module, \( \mu \) is equal to the receiving rate at the wireless STA. In Eq. (21), values of \( T_a \) and \( T_r \) can be calculated via Eq. (1) and Eq. (2). For LL attack, ML attack and HL attack, the UDP packet loss is expressed as Eq. (23), Eq. (24) and Eq. (25), respectively.

\[ \text{PL}(\%)=0 \]  
(23)  
\[ \text{PL}(\%)=100(1- (1-S_3-1/\lambda_3)/(S_3S_2\lambda_3)) \]  
(24)  
\[ \text{PL}(\%) = 100 \]  
(25)  

VII. SIMULATION AND RESULT DISCUSSIONS

A. Development of simulation for the queuing model

The simulation components for the WLAN queuing model (Fig. 3) include the frame generator, the frame receiver, and the frame handler. Frame generators include the wireless STA, the wired STA, and the hacker STA. Frame receivers are the RX queue, TX1 queue, TX2 queue, and end user frame receiver array (array EUFR). We define array EUFR to hold user frames. Frame handlers include the AP_Air RX (AAR) service center, and the AP_Air TX service center. The
simulation model is built with Java thread programming. For the simulation presented in this paper, the data frames and attacking frames are generated with a constant rate. The simulation program can be easily enhanced to handle randomly (e.g. Poisson distributed) generated data frames and attacking frames. The simulation threads include the frame generator (D11FrameGenerator) thread, the first frame handler (D11FrameHandler-1) thread, and the second frame handler (D11FrameHandler-2) thread. Based on the WLAN queuing model, we create the simulation diagram as Fig. 5. It shows the interactions among D11FrameGenerator, queue_AAR, D11FrameHandler-1, D11FrameHandler-2, queue_TX1, queue_TX2, and array_EUFR.

For the wireless STA and the attacker, the D11FrameGenerator thread produces management frames and data frames, and adds them into the RX queue (queue_AAR). For the wired STA, D11FrameGenerator produces data frames and adds them into the queue_TX2. D11FrameHandler-1 thread behaves as the AP_Air RX service center and periodically retrieves 802.11 frames from the queue_AAR and adds them into queue_TX1 or queue_TX2. D11FrameHandler-2 thread represents AP_Air TX; it retrieves frames from the queue_TX1, or the queue_TX2, and adds them into array_EUFR. D11FrameHandler-2 has a higher priority to process frames from the queue_TX1. When an 802.11 frame is added into a queue, the current timestamp is appended to that frame. TCP RTT or UDP packet loss is calculated based on the received data frames inside the array_EUFR.

B. Results and discussions
In this section, the analytical and simulation results under AuthRF DoS attacks are discussed. Similar discussions could also be applied to AssRF DoS attacks, which are not included in this paper.

The upstream UDP packet losses under different attacking rates are shown in Fig. 6. The data from the experiment also matches the results from the analytical model and simulation, except for a minor outlier (fps=5.0) from the experiment.

Comparisons of empirical, simulation, and analytical results for TCP RTT are shown in Fig. 7. Based on Eq. (15), Eq. (16), and Eq. (17), TCP RTT is calculated under different AuthRF attacking rates. Meanwhile, TCP RTT is also obtained from the simulation model under different AuthRF attacking rates. It shows that the data from the experiment matches the results from both the analytical model and the simulation.

The downstream UDP packet losses under different attacking rates are shown in Fig. 8. The data from the experiment also matches the results from the analytical model and the simulation, except for a minor deviation at fps=8.0.
Implementing any of these approaches will enable the AP to deauthenticate and disassociation frames. This is similar to the method for authenticating bit authentication and Message Integrity Check (MIC), Address Filtering (MAF), sequence number, and random request frames. Examples of RA mechanisms include MAC authenticity of incoming authentication and association process data frames. RA requires the AP to validate the TX1 queue. Thus, the TX2 queue will have time to prepare and transmit response frames to the fake wireless STA. Thus, the AP will not stop processing the fake requests. In our research, we implemented MAF to demonstrate the effectiveness of the RA solution.

The open source based HostAP’s driver is modified to implement the MAF authentication algorithm to stop processing fake 802.11 request frames. The relationships between TCP RTTs and attacking rates are presented in Table IX.

### VII. APPROACHES TO RESOLVE DOS ATTACKS

Analytical, simulation, and experimental data show that the performance impact of AuthRF/AssRF is related to the attack duration (D), the attacking rate (λ), and the average processing time (S2) for a fake 802.11 request. The hacker controls the attack duration and the attacking rate. The fake request processing time of S2 is related to the 802.11 implementations inside the AP. S2 is composed of Srep, N1, and N2, which are defined as follows:

1. Srep refers to the service time for the AP_Air TX (TX1) to send out one 802.11 response frame.
2. N1 refers to the number of transmitted 802.11 response frames related to one 802.11 request frame. Currently, the 802.11 standard specifies N1 with a default value of 7.
3. N2 refers to the number of copies of the same 802.11 request processed by the AP. Under the current experimental environment, N2 has a value of 3.

The relationship among S2, Srep, N1, and N2 is expressed in Eq. (26).

\[ S_2 = S_{rep} \times N_1 \times N_2 \]  

(26)

To resolve AuthRF and AssRF attacks, four approaches are proposed and validated with simulation, analytical, and experimental data.

#### A. Request Authentication (RA)

Our first approach to address the issue is to reduce the attacking rate by allowing AP to discard faked requests. Currently, 802.11 does not require authentication of AuthRF/AssRF requests. RA requires the AP to validate the authenticity of incoming authentication and association request frames. Examples of RA mechanisms include MAC Address Filtering (MAF), sequence number, random bit authentication, and Message Integrity Check (MIC), which is similar to the method for authenticating deauthentication and disassociation frames. Implementing any of these approaches will enable the AP to stop processing fake request frames. In our research, we implemented MAF to demonstrate the effectiveness of the RA solution.

The open source based HostAP’s driver is modified to implement the MAF authentication algorithm to stop processing fake 802.11 request frames. The relationships between TCP RTTs and attacking rates are presented in Table IX.

<table>
<thead>
<tr>
<th>Attacking Rate (fps)</th>
<th>1</th>
<th>3</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline RTT data (ms)</td>
<td>15</td>
<td>26</td>
<td>47</td>
<td>47</td>
<td>464</td>
<td>5190</td>
</tr>
<tr>
<td>RA empirical data (ms)</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>RA analytical data (ms)</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>RA simulation data (ms)</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
</tr>
</tbody>
</table>

#### B. Reduction of Duplicate Requests (RDR)

In the case of the Linksys WAP54G AP, the experimental data shows that three copies of the same fake authentication request frame leads to the transmission of 21 response frames to the fake wireless STA. If the AP implements a mechanism to stop processing duplicate frames, it will reduce the number of transmissions to the fake wireless STA. As described in Eq. (26), reducing N2 will decrease S2, which in turn reduces the DoS effects.

We implemented RDR on a Compaq WL200 HostAP. With RDR, the value of N2 in Eq. (26) is set to 1, and the relationship of TCP RTT vs. attacking rates is shown in Table X. The experimental, simulation, and analytical results show that RDR significantly reduces the effect of the attack. With the implementation of RDR, an attacking rate of 8.0 fps only leads to TCP RTT values less than 25 ms.

<table>
<thead>
<tr>
<th>Attacking Rate (fps)</th>
<th>1</th>
<th>3</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline RTT data (ms)</td>
<td>15</td>
<td>26</td>
<td>47</td>
<td>47</td>
<td>464</td>
<td>5190</td>
</tr>
<tr>
<td>RDR empirical data (ms)</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td>17</td>
<td>17</td>
<td>17</td>
</tr>
<tr>
<td>RDR analytical data (ms)</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td>17</td>
<td>18</td>
<td>20</td>
</tr>
<tr>
<td>RDR simulation data (ms)</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td>16</td>
<td>16</td>
<td>17</td>
</tr>
</tbody>
</table>

#### C. Reduction of Response Retransmission (RRR)

Equation (26) shows that S2 is also closely related to the (re)transmission threshold (N1). Reducing N1 from 7 to a smaller number like 1 will reduce the value of S2. In our research, the Netgear MA401 HostAP is used to implement the RRR solution. The relationship between TCP RTT and attacking rates is shown in Table XI. With the implementation of RRR, RTT values increases by only a small margin. The tradeoff of RRR is that the lost valid response frames are not retransmitted.

<table>
<thead>
<tr>
<th>Attacking Rate (fps)</th>
<th>1</th>
<th>3</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline RTT data (ms)</td>
<td>15</td>
<td>26</td>
<td>47</td>
<td>47</td>
<td>464</td>
<td>5190</td>
</tr>
<tr>
<td>RRR empirical data (ms)</td>
<td>10</td>
<td>10</td>
<td>12</td>
<td>10</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>RRR analytical data (ms)</td>
<td>10</td>
<td>10</td>
<td>12</td>
<td>10</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>RRR simulation data (ms)</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

#### D. Round Robin Transmission (RRT)

Currently, the TX1 queue has preemptive priority over the TX2 queue. To resolve DoS attack effects, we implement the round robin transmission (RRT) mechanism and ensure that...
TX1 and TX2 queues have the same chance to transmit the data frames and management frames. For example, after the TX1 transmits one management frame, the TX2 will transmit one data frame, which is called RRT-1/1.

To enable the AP to transmit management and data frames according to the RRT approach, the firmware of the AP wireless NIC needs to be modified. However, the open source HostAP and the commercial APs vendors do not provide options for modifying its firmware. Thus, in our research, we only provide analytical and simulation data to verify RRT.

With an implementation of RRT-1/1, the simulation and analytical results are shown in Table XII. Under the protection of RRT, a high attacking rate of 8.0 fps only slightly increases RTT values (≤13 ms). The tradeoff of RRT is the slower response to management frames.

<table>
<thead>
<tr>
<th>Attacking Rate (fps)</th>
<th>1</th>
<th>3</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline RTT data (ms)</td>
<td>15</td>
<td>26</td>
<td>47</td>
<td>47</td>
<td>464</td>
<td>5190</td>
</tr>
<tr>
<td>RRT analytical data (ms)</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>RRT simulation data (ms)</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>13</td>
<td>13</td>
</tr>
</tbody>
</table>

### E. Comparisons of RA, RDR, RRR and RRT

As illustrated in Fig. 9, four approaches of RA, RDR, RRR, and RRT can effectively reduce the RTT value from 5190 ms to a value less than 20 ms for attacking rate 8 fps.

Among these approaches, RA seems to be the most effective, as it can resolve AuthRF/AssRF DoS attacks regardless of the attacking rate. However, the implementation of RA needs to modify current 802.11 frame format to convey additional information for the AP to verify the authenticity of authentication or association request frames. It also needs a central database to contain wireless STA information for the AP to verify authentication/association request frames. In this paper, we have a simple implementation of RA via MAF. The shortcoming of MAF is that MAC address spoofing will defeat it. Other more complex and secure RA approaches include sequence number [24], random bit authentication [25], and MIC [12], which can defeat MAC address spoofing.

AuthorRF with an attacking rate of 50 fps still stops TCP data communications.

The implementation of RRT does not need to change 802.11 MAC frame format, and it only needs to change the algorithms used by the AP wireless NIC to transmit management frames and data frames. With the implementation of RRT-1/1, in the worst-case scenario of a high attacking rate, we will observe 50% performance degradation. RRT is less effective than RA, but more effective than RDR and RRR at high attacking rates.

In summary, RA is the best solution to resolve AuthRF and AssRF DoS attacks. However, authentications of 802.11 authentication/association request frames are not specified in 802.11 standards. Before RA becomes a part of 802.11 standards, AP vendors could adopt RDR, RRR, and/or RRT to reduce the negative effects of AuthRF and AssRF attacks.

### IX. CONCLUSIONS

In this paper, we demonstrate AuthRF and AssRF DoS attacks and their effects on WLAN communications. An AP based WLAN queuing model is proposed to analyze TCP/UDP traffic under these attacks. Simulation and analytical results of our queuing model match experimental results. Our study shows that current implementations of 802.11 have some limitations, which lead to the transmission of too many response frames. Four approaches are proposed to address this issue. The empirical and analytical results show that they are all effective in resolving the AuthRF and AssRF attacks performance degradation, and the pros and cons are primarily in the ease of implementation.

In our research, we also studied the effects of AuthRF and AssRF on wireless voice over IP (WVoIP). Since VoIP has symmetric UDP data flow of upstream and downstream traffic, AuthRF and AssRF always cause serious problems to VoIP. The same solutions can also be used to resolve AuthRF and AssRF attacks over WVoIP.

### REFERENCES


