2019 27th Mediterranean Conference on Control and Automation (MED)
July 1–4, 2019 · Akko · Israel

Evil Twin Attack Detection using Discrete Event Systems in IEEE 802.11 Wi-Fi Networks

Nirmal S Selvarathinam, Amit Kumar Dhar and Santosh Biswas*
Department of EECS, Indian Institute of Technology Bhilai
{nirmals,amitkdhar,santosh}@iitbhilai.ac.in

***Abstract*— Wi-Fi technology has seen rapid growth in the last two decades. It has revolutionized the way we access the Internet. However, they are vulnerable to Denial of Service attacks, Encryption Cracking, and Rogue Access Points etc. In this manuscript, our focus is on *Evil Twin Attack*, the most common type of Rogue Access Point (RAP). An evil twin AP lures client(s) into connecting to it, disguising itself as a genuine AP by spoofing its MAC address and SSID (Service Set IDentifier). Once a client is connected to the evil twin AP, the attacker can spy on its communication, re-direct client(s) to malicious websites, compromise credentials. Whitelisting AP(s), timing based solutions, patching AP/client etc., are some existing methods to detect evil twin AP(s) in a network. However, practically methods demand comprehensive set up and maintenance, they suffer from scalability and compatibility issues. Some even require protocol modifications, thus making it expensive and practically infeasible in a large scale network with no proof of correctness. To address these issues, we propose a Discrete Event System (DES) based approach for Intrusion Detection System (IDS) for evil twin attacks in a Wi-Fi network.**

I. Introduction

Wireless technologies such as Wi-Fi, Bluetooth, Infrared, Radio, Microwave, Zigbee etc [12], [10], [1] have seen enormous advancement in terms of its applications. Notably, Wi-Fi is one of the fastest growing technologies in the world. As communication through Wi-Fi takes place over the air, it can be tapped easily within the range of the corresponding Access Point (AP). Wi-Fi networks are vulnerable to a series of attacks like jamming[14], man-in-the-middle [9], MAC spoofing [6], authentication flood, de-authentication flood etc.

A Rogue Access Point (RAP) is the most common and effective way to compromise Wi-Fi networks. An evil twin AP is a RAP that replicates the MAC address and Service Set IDentifier (SSID) of a genuine AP[1], thus creating an evil twin in the network. As a result, when a new client/device connects to that network, it can see a list of available Wi-Fi APs with more than one AP having the same SSID. Many of the modern operating systems (OS) are configured to hook up with the AP that offers the highest Received Signal Strength Indication (RSSI).

Fig. 1a, describes the normal scenario where there is no RAP(s) within the network. Now, in case of an evil twin AP being present, leads to two possibilites. Either the evil twin AP could bridge the connection via genuine AP for Internet access, as shown in Fig .1b or it may provide a private Internet connection for the clients, as shown in Fig .1b. Hence, we need a robust solution that is capable of detecting the presence of evil twin regardless of the type of Internet access, the attacker chooses to provide.

There are various techniques proposed in the literature to detect the presence of evil twin AP(s) in a Wi-Fi network. Some such schemes are: *Traffic Monitoring* [16] and Chirumamilla et al. [7], *Timing Based Solutions* Mano et al. [13] [15] [11], *Special Hardware Solutions* [8], [2], *Statistical Evaluation* [5] etc. All these techniques have their own drawbacks. Monitoring wireless traffic by deploying sniffers in a network becomes cumbersome and difficult to manage. It may also be fallible in cases where MAC addresses are spoofed to conceal the identity of the evil twin AP. Timing based techniques with additional hop count do not work when the evil twin AP uses its own private internet connection for serving clients' requests. Statistical evaluation techniques suffer from a large false alarm rate. In [4] Mayank et al., have proposed a better approach compared to the schemes discussed above. It proposes an algorithm that checks for the presence of evil twin AP(s) in a network using some key fields from authentication and association frames. But there is no guarantee of correctness in its proposed solution. It is strictly an ad-hoc approach to detecting evil twin AP(s) in a Wi-Fi network.

From the review, it appears that we require a method that is not merely a technique to solve the evil twin AP problem but also a way of proving its correctness. In this manuscript, we propose a Discrete Event System(DES) based Intrusion Detection System(IDS). DES has successfully been used for failure detection and diagnosis of large systems like nuclear reactors, power plants etc. The key idea here is to design a DES model for the system under normal and attack scenarios, so that our system could be further extended with proof of correctness. A detector is a state estimator of the model, that keeps track of the system using events and alert if the system moves into an attack state. The idea of DES based IDS has already been established in [3] in wired networks. Such IDS provides better accuracy and detection rate. We are adopting the DES based IDS for Wi-Fi networks in this manuscript.

II. Background: Evil Twin Attack

Fig. 2 demonstrates evil twin attack using timeline. Here we assume an open Wi-Fi network. Y is a genuine AP, X

[1]An AP set up by network administrator is referred to as Genuine AP.

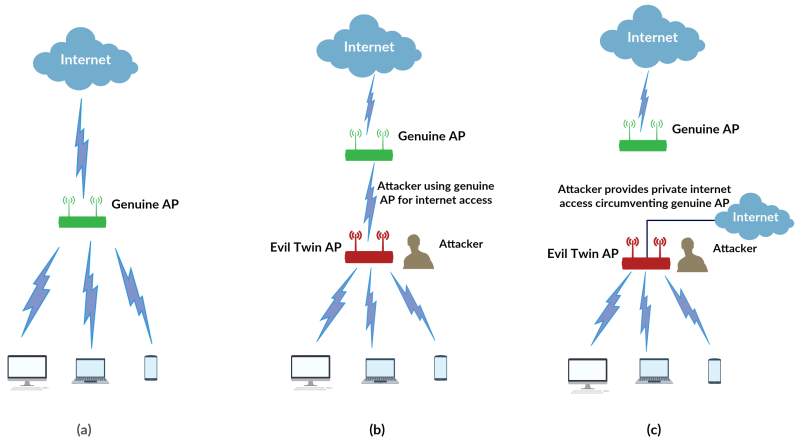

Fig. 1: Normal and Evil Twin Setup.

is an evil twin AP in the network. The attacker follows the timeline as demonstrated in Fig. 2 to execute evil twin attack. Each of the distinct time-slot shown in Fig. 2 is explained as follows:

1) $[1^{st}\ Timeslot\ (T_1) - Handshake(1/4)]$: A Client sends an authentication request frame to Y, X is dormant and sniffs all the frames exchanged in the network silently.
2) $[2^{nd}\ Timeslot\ (T_2) - Handshake(2/4)]$: Y responds with an authentication response frame. An open Wi-Fi network does not require any key exchange for authentication, instead it requires a simple frame exchange between the client and the AP. X silently maintains a list of clients and their respective parameter information authenticated to Y without revealing its presence to anyone in the network. It operates in complete stealth mode preparing for the association phase.
3) $[3^{rd}\ Timeslot\ (T_3) - Handshake(3/4)]$: The Client sends an association request frame to Y.
4) $[4^{th}\ Timeslot\ (T_4) - Handshake(4/4)]$: X sends an association response frame to the Client before Y.
5) $[5^{th}\ Timeslot\ (T_5)]$: Y too sniffs an association request frame sent by the Client in time-slot (T_3). Thereby Y, sends an association response frame.
6) $[6^{th}\ Timeslot\ (T_6)]$: From this point onwards the Client is associated with X instead of Y for data communication, as X's response was received first. The Client drops the association response frame sent by Y.

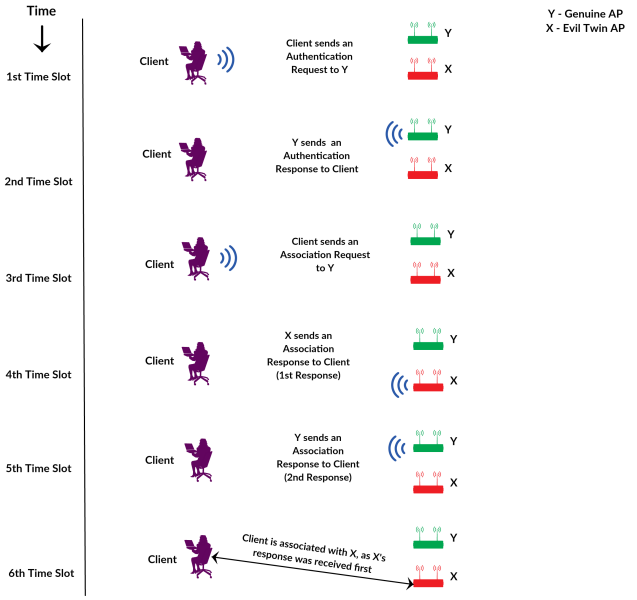

Fig. 2: Evil Twin Attack Timeline

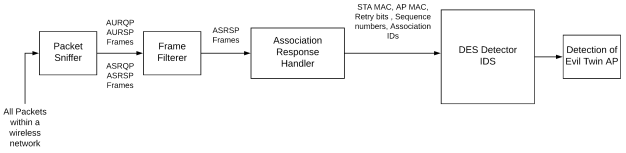

Fig. 3: Block diagram of implementation of IDS

III. Proposed Scheme

In this section, we propose a Discrete Event System(DES) based Intrusion Detection Scheme(IDS) for detecting evil twin attacks in Wi-Fi networks. The following assumptions are made in 802.11 Wi- Fi networks.

1) Evil twin AP(s) choose not to send beacon frames and not to reply to probe requests.
2) Evil twin AP(s) could circumvent genuine AP by providing a private internet connection of its own.
3) A genuine AP supposedly responds to client's requests.
4) The IDS is restricted to montior only the AP(s) authorized by the network administrator.
5) The IDS has sufficient sniffing capabilities.

The essential idea here is to use retry bit, sequence number and association ID fields of association response frame sent from an AP to a client. By examining these key fields the presence of evil twin AP(s) in a Wi-Fi network can

be detected [4]. In this manuscript, we use this concept by introducing DES models for Wi-Fi networks for both normal and attack scenarios. Any DES model has states and transitions. A transition is made from one state to another based on the occurrence of some discrete event(s). These events are modeled on the basis of certain changes in the system. In this case, we consider the following changes in the network as events: Sending/receiving an authentication frame, sending/receiving an association frame etc.

As shown in Fig. 3 all packets in a Wi-Fi network are sniffed with a packet sniffer module. The frame filterer module captures only the association response frame. These association response frames are given as inputs to an association response handler. An ASSOCIATION-RESPONSE-HANDLER is formalized as follows:

ASSOCIATION-RESPONSE-HANDLER
Input: $ASRSP$ frame
Output: Pass R_1, R_2, Seq_1, Seq_2, AID_1, AID_2 to the IDS module
IF($PRQP$ frame is malformed)
"Status is malformed" and EXIT
END IF

The role of ASSOCIATION-RESPONSE-HANDLER is to acknowledge receipt of $ASRSP$ frames; Send R_1, R_2, Seq_1, Seq_2, AID_1, AID_2 to the DES detector module.

The DES detector is discussed in the next section of our manuscript.

The following are some events and their short notations used in this manuscript:
$AURQP$ - Authentication Request;
$AURSP$ - Authentication Response;
$ASRQP$ - Association Request;
$ASRSP$ - Association Response;
$STA\ MAC$ - Station/Client MAC Address;
$AP\ MAC$ - Access Point MAC Address;
DF - Data Frame;
GAP - Genuine Access Point;
RAP - Rogue Access Point;
$BSSID\ GAP$ - Basic Service Set IDentifier of a GAP;
$BSSID\ RAP$ - Basic Service Set IDentifier of a RAP;
R_1 - Retry bit of 1st Association Response($ASRSP_1$);
R_2 - Retry bit of 2nd Association Response($ASRSP_2$);
Seq_1 - Sequence number of ($ASRSP_1$);
Seq_2 - Sequence number of ($ASRSP_2$);
AID_1 - Association ID of ($ASRSP_1$);
AID_2 - Association ID of ($ASRSP_2$);
$PRQP$ - Probe Request Frame;
$PRSP$ - Probe Response Frame;
$MACC$ - Client MAC Address;
$MACAP$ - Access Point MAC Address;

A. DES Modeling

The DES model used for representing Wi-Fi networks under normal & attack conditions is a 5 tuple $\langle \Sigma, S, S_0, V, \Im \rangle$, where Σ is the set of events, S is the set of states, $S_0 \subseteq S$ is the set of initial states, V is the set of model variables and $\Im$ is the set of transitions. It may be noted that there is no final state as $AURQP$, $AURSP$, $ASRQP$, $ASRSP$ frames are always present in a wireless network as there might be new stations(STA) trying to connect to the network.

$\Sigma = \{AURQP, AURSP, ASRQP, ASRSP, DF\}$. The set of states S is shown in Fig. 4 and Fig. 5. States with no primes correspond to normal scenario and those with primes denote attack scenario. Initial states (S_0) are also shown in Fig. 4 and Fig. 5. Model variable set is $V = \{MACC, MACAP\}$; $MACC$ & $MACAP$ has the domain as $D_2 = \{hh-hh-hh-hh-hh-hh|h \in Hex\}$.

A transition $\tau \in \Im$ is a five-tuple $\langle s, s', \sigma, \phi(V), Assign(V) \rangle$, where s is the source state, s' is the destination state, σ is an event (on which a transition happens), $\phi(V)$ is boolean conjunction of equalities of a subset of variables in V, and $Assign(V)$ is a subset of model variables and assignments with values from their corresponding domains. It may be noted that there are "$-$" for some fields in the tuple representing the transitions. if the "$-$" is for $\phi(V)$ or $Assign(V)$ then it represents NO action (i.e., reset or assignment) is required.

1) DES in Normal Mode: An overview of DES model under normal conditions as shown in Fig. 4 is as follows: Initially, the system is in state $S1$, a transition from $S1$ to $S2$ takes place when there is an authentication request frame sent by a client/STA to an AP. In this transition the model variables $MACC$ and $MACAP$ are initialized by $STA\ MAC$ address and $AP\ MAC$ address respectively. The system moves from state $S2$ to $S3$ when there is an authentication response frame received by the client/STA from the respective AP it wants to connect to. In this transition, we verify if $MACC$ is the same as the $STA\ MAC$ address and $MACAP$ is same as the respective $AP\ MAC$ address. There are no new initializations in this transition. A transition occurs from state $S3$ to $S4$ in the event of an association request frame sent by the client/STA to the AP. Similar model variable condition checks are done and no new initializations are followed. The system moves from state $S4$ to $S5$ when the client/STA receives its first association response frame. Since the system is not certain if the association response frame is in fact sent by a genuine AP, it stores retry bit, sequence number and association ID in its model variables R_1, S_1 and AID_1 respectively. A transition from state $S5$ to $S6$ or $S5$ to $S7$ indicates possible cases of re-transmissions attempted by the genuine AP considering the 1st association response frame is lost. A transition from state $S5$ to $S6$ has R_1 = 0 and R_2 = 1, this represents a possible case of frame retransmission attempted by a genuine AP. Furthermore, the condition check Seq_1 = Seq_2 and AID_1 = AID_2 ensures the fact that it is a case of frame retransmission by attempted a genuine AP. Similarly, a transition from $S5$ to $S7$ has R_1 = 1 and R_2 = 1, which is verified for a genuine case of retransmission by checking the sequence number and association ID fields i.e., Seq_1 = Seq_2 and AID_1 = AID_2.

2) DES in Attack Mode: The DES model under attack conditions as shown in Fig. 5 works as follows: A transition

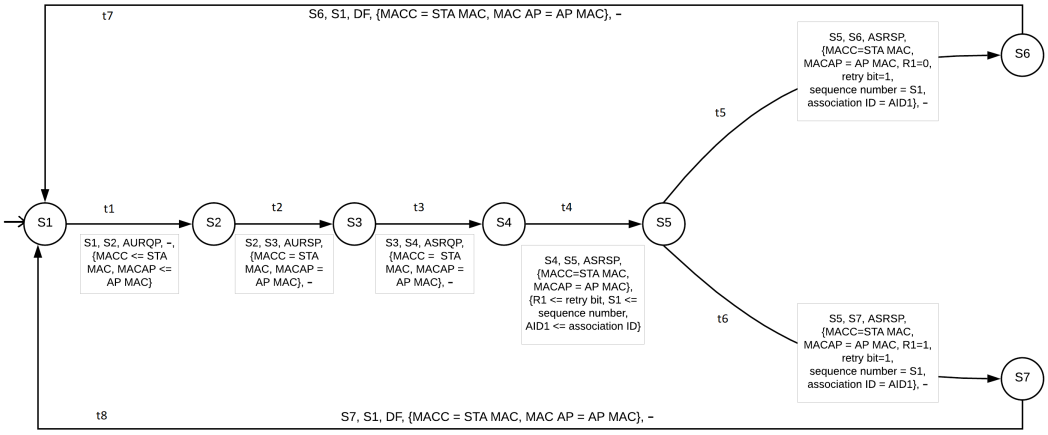

Fig. 4: DES based model under normal conditions

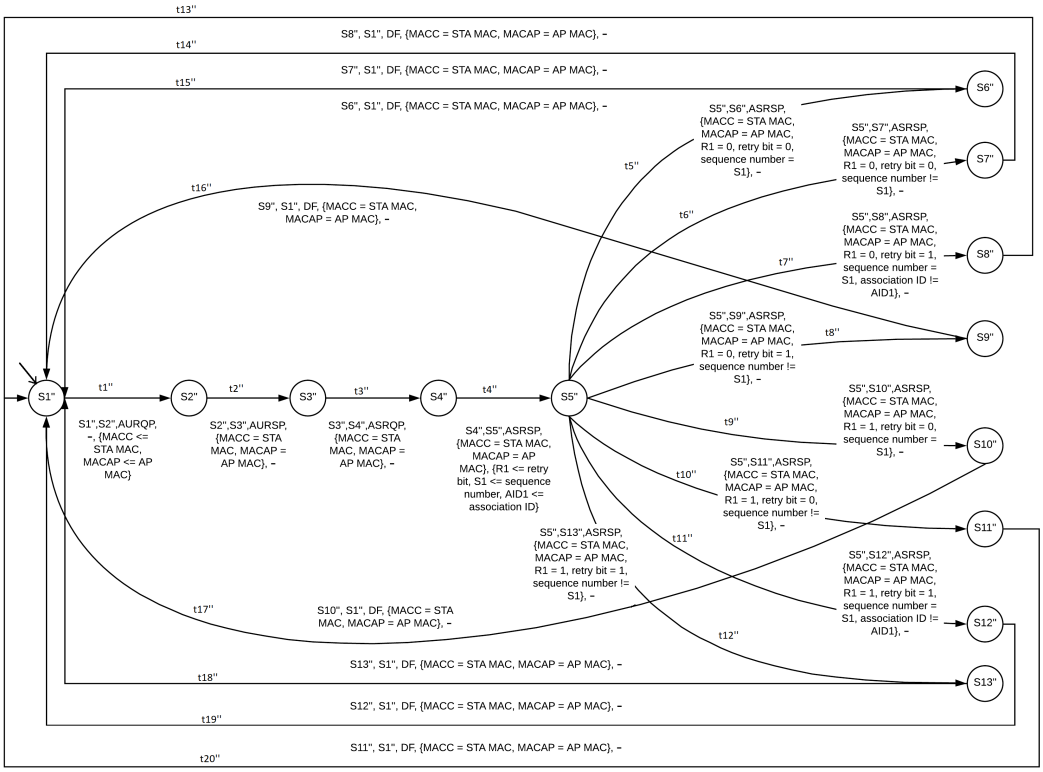

Fig. 5: DES based model under attack conditions

from $S1''$ to $S2''$ takes place when there is an authentication request frame sent by a client/STA to an AP. In this transition the model variables $MACC$ and $MACAP$ are initialized by $STA\ MAC$ address and $AP\ MAC$ address respectively. The system moves from state $S2''$ to $S3''$ when there is an authentication response frame received by the client/STA from the respective AP it wants to connect to. In this transition, it is checked if $MACC$ is same as the $STA\ MAC$ address and $MACAP$ is same as the respective $AP\ MAC$ address. There are no new initializations in this transition. A transition occurs from state $S3''$ to $S4''$ in the event of an association request frame sent by the client/STA to the AP. Similar model variable condition checks are done and no new initializations are followed. The system moves from state $S4''$ to $S5''$ when the client/STA receives its first association response frame. Since the system is not certain if the association response frame is in fact sent by a genuine AP, it stores retry bit, sequence number and association ID in its model variables R_1, S_1 and AID_1 respectively. The system moves from state $S5''$ to $S6''$ when R_1 = 0, R_2 = 0 and Seq_1 = Seq_2. The same sequence number indicates the possibility of retransmission attempted by the genuine AP.

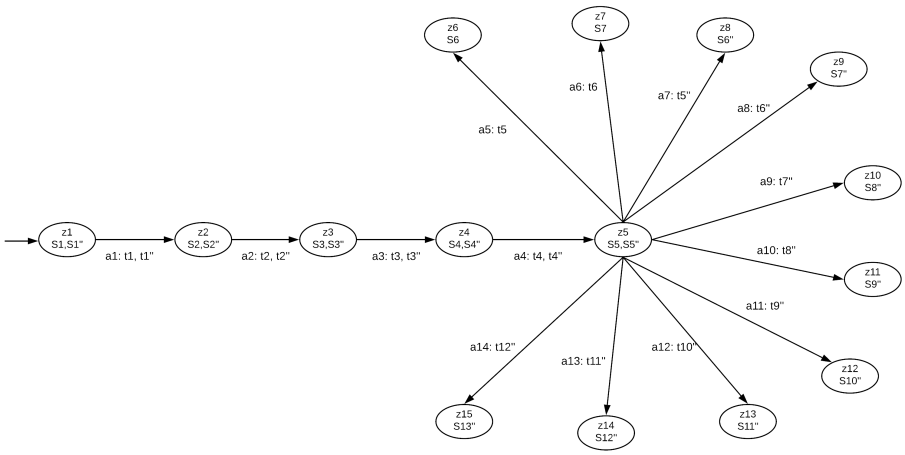

Fig. 6: Detector of DES model

Since R_2 is set to 0, this case is only possible if there is an evil twin AP in the network. In a similar fashion, other transitions originating from state $S5$ could be explained.

B. Detector

A detector is primarily a state estimator of the DES model under consideration. It keeps track of the system using events and alerts if the system moves into an attack state. Once the system terminates in a normal or attack certain state, the frame details are recorded accordingly to conclude the presence or absence of an evil twin AP in a network.[3]

Exposition 1 *Conditions on Equivalent States and Transitions*: Assume that the DES corresponding to the normal condition and attack condition are represented as : $\langle \Sigma, S, S_0, V, \Im \rangle$, $\langle \Sigma, S', S'_0, V, \Im' \rangle$, respectively. Two transitions $\tau_1 = \langle s_1, s_2, \sigma_1, \phi_1(V), Assign_1(V) \rangle \in \Im$ and $\tau_2 = \langle s'_1, s'_2, \sigma_2, \phi_2(V), Assign_2(V) \rangle \in \Im'$ are said to be equivalent if they have same events, equalities and assignment values. In notations: $\sigma_1 = \sigma_2$, $\phi_1(V) = \phi_2(V)$ and $Assign_1(V) = Assign_2(V)$.

If $\tau_1 = \tau_2$, then the corresponding source states s_1 and s_2 are considered the same. Similarly, the corresponding destination states s'_1 and s'_2 are considered the same. Thus, these similar states are merged into one state.

The DES detector could be represented as a directed graph $O = \langle Z, A, Z_0 \rangle$, where Z is a set of detector states, known as *O-states* and A is a set of detector transitions, known as *O-transitions* and Z_0 is the set of initial O-states. The states of the detector is defined as following.

- $Z = S \cup S' \cup S \times S'$
- $Z_0 = S_0 \times S'_0$

The set of transitions A is defined as union of transitions of the form $\langle (s_1, s'_1), (s_2, s'_2) \rangle$ for all transitions such that $\langle s_1, s_2, \sigma_1, \phi_1(V), Assign_1(V) \rangle = \langle s'_1, s'_2, \sigma_2, \phi_2(V), Assign_2(V) \rangle \}$ and transitions of the form $\langle (s_1, s'_1), (s'_2) \rangle, \langle (s_1, s'_1), (s_2) \rangle$ for all transitions such that $\langle s_1, s_2, \sigma_1, \phi_1(V), Assign_1(V) \rangle \neq \langle s'_1, s'_2, \sigma_2, \phi_2(V), Assign_2(V) \rangle$.

The initial O-state is a combination of initial states of both DES models(attack and normal). Each O-state is a subset of equivalent model states from both attack and normal behaviour models $z \in Z$. The O-transitions originating from z are nothing but a combination of equivalent transitions from the DES models (normal and attack models) from one set of equivalent source states to another set of equivalent destination states.

Exposition 2 *Normal Certain O-State*: An O-state that exclusively contains state(s) from the normal behaviour model.

Exposition 3 *Attack Certain O-State*: An O-state that exclusively contains state(s) from the attack behaviour model.

Fig. 6 shows the detector corresponding to our DES models under consideration Fig. 4 and Fig.5. The detector works as follows:

1) The initial state of the detector Z1 contains all initial states from all models i.e., S1, S1".
2) Let $\Im_{z1} = \{t1, t1"\}$ i.e., all the outgoing model transitions from the model states in $z1 = \{S1, S1"\}$. As all transitions in $\Im_{z1}$ are equivalent, $\Im_{z1}$ cannot be further partitioned into equivalent classes. Thus, a_1 is an O-transition corresponding to $\{t1, t1"\}$.
3) The destination O-state to the O-transition a_1 is $z2 = \{S2, S2"\}$, as the destination model state for $t1$ and $t1"$ is $S2$ and $S2"$ respectively.
4) The states $z8$, $z9$, $z10$, $z11$, $z12$, $z13$, $z14$ & $z15$ are known as attack certain states. Similarly, states $z6$ & $z7$ are known as normal certain states. This is because all these states exclusively contain either attack model state(s) or normal model state(s) from our DES models.

The normal or attack scenario detection is done by determining if the system reaches any of the normal or attack certain states respectively. This holds good irrespective of the sequence of O-states and O-transitions followed before reaching a normal certain or attack certain state. For instance, if the system arrives at state $z11$ by following $z1$, $z2$, $z3$, $z4$ and $z5$, an attack is detected since $z11$ is an attack certain state. Likewise, if the system reaches the state $z6$ following

TABLE I: Detection Rate analysis using the proposed IDS

Trial	# of attack instances launched	# of instances detected successfully by proposed IDS	Detection Rate %	Detection Rate % [4]
1	100	94	94.00	92.50
2	100	93	93.00	93.33
3	100	91	91.00	95.83
4	100	94	94.00	98.33
5	100	100	100.00	95.00
6	100	96	96.00	99.17
7	100	96	96.00	95.00
8	100	94	94.00	100.00
9	100	94	94.00	97.50
10	100	95	95.00	95.00

$z1$, $z2$, $z3$, $z4$ and $z5$ assures a normal behaviour of the system, as $z6$ is a normal certain state.

IV. Implementation, experimental results and comparison

Our results have been tabulated in Table I. An experimental local network was set up with a genuine access point with SSID 'Coffee Shop' and MAC address '94:65:2d:ce:76:e3'. An evil twin AP was configured on a machine with Kali Linux operating system. The IDS was implemented in C, running on an Ubuntu 16.04 machine placed in close proximity with the genuine AP to obtain the best possible results. Since the evil twin AP has spoofed its SSID and MAC address with the genuine AP, a new user who enters our network has to connect to one of the 'Coffee Shop' listed. For our experiment, we used a few Android and iOS smartphones for testing the IDS. The key information needed for detecting the presence or absence of evil twin AP(s) in the network was acquired from the authentication and association frames that were captured by our IDS. Aircrack-ng suite was used for creating the evil twin AP(s). The key fields from these frames were extracted and stored in the local MySQL database running on the IDS machine. MySQL database only maintains key information fields such as retry bits, sequence number, association ID etc., of the association response frames captured. Our IDS program retrieves essential fields from the MySQL database to comprehensively detect the presence or absence of evil twin AP(s) based on the equality conditions applied on key fields of the two association response frames with same SSID and MAC address. By optimizing the IDS we obtained a slightly better detection rate of 94.7% on average.

Ideally, the detection rate is expected to be 100%. However, in practise some association response frames are not captured by the IDS, as a wireless medium is noisy. When the IDS fails to capture one of the two association response frames, it does not consider detecting the possibility of presence of evil twin AP(s) in the network. The detection rate varies from 93% to 100%.

V. Conclusion

In this manuscript, we propose a DES detector based IDS for detecting evil twin AP(s) in Wi-Fi networks. This scheme can detect the presence of evil twin AP(s) in Wi-Fi networks with an accuracy of 94.7% on average. Furthermore, the proposed scheme is easy to implement, no protocol modification is required, low network overhead and offers proof of correctness.

References

[1] Wireless Infrared Communication Systems and Networks. International Journal of Wireless Information Networks **4**(4), 257–258 (1997)

[2] AirWave Wireless Management Suite, Whitepaper, Aruba (2006). URL www.moonblinkwifi.com/files/airwave-solution-guide.pdf

[3] Neminath Hubballi, Santosh Biswas, S Roopa, Ritesh Ratti, Sukumar Nandi: LAN attack detection using Discrete Event Systems: ISA Transactions: Volume 50, Issue 1: Pages 119-130 (2011)

[4] Mayank Agarwal, Santosh Biswas, Sukumar Nandi: An Efficient Scheme to Detect Evil Twin Rogue Access Point Attack in 802.11 Wi-Fi Networks: International Journal of Wireless Information Networks: Volume 25: Issue 2, pp 130-145 (2018)

[5] Agrawal, N., Tapaswi, S.: The Performance Analysis of Honeypot Based Intrusion Detection System for Wireless Network. International Journal of Wireless Information Networks **24**(1), 14–26 (2017)

[6] Bellardo, J., Savage, S.: 802.11 denial-of-service attacks: real vulnerabilities and practical solutions. In: Proceedings of the 12th conference on USENIX Security Symposium - Volume 12, SSYM'03, pp. 15–28 (2003)

[7] Chirumamilla, M.K.: Agent based Intrusion Detection and Response system for Wireless LANs. In: Proceedings of IEEE International Conference on Communications, pp. 492–496 (2003)

[8] Dubey, P.K., Verma, J.N.: Method And Apparatus For Detecting A Rogue Access Point In A Communication Network (2012). URL http://www.google.com/patents/US20120124665

[9] Gilad, Y., Herzberg, A.: Off-path tcp injection attacks. ACM Transactions on Information and System Security (TISSEC) **16**(4), 13:1–13:32 (2014)

[10] Gutiérrez, J.A.: On the Use of IEEE Std. 802.15.4 to Enable Wireless Sensor Networks in Building Automation. International Journal of Wireless Information Networks **14**(4), 295–301 (2007)

[11] Han, H., Sheng, B., Tan, C., Li, Q., Lu, S.: A Timing-Based Scheme for Rogue AP Detection. IEEE Transactions on Parallel and Distributed Systems **22**(11), 1912–1925 (2011)

[12] Johansson, P., Kapoor, R., Kazantzidis, M., Gerla, M.: Personal Area Networks: Bluetooth or IEEE 802.11? International Journal of Wireless Information Networks **9**(2), 89–103 (2002)

[13] Mano, C.D., Blaich, A., Liao, Q., Jiang, Y., Cieslak, D.A., Salyers, D.C., Striegel, A.: RIPPS: Rogue Identifying Packet Payload Slicer Detecting Unauthorized Wireless Hosts Through Network Traffic Conditioning. ACM Trans. Inf. Syst. Secur. **11**(2), 2:1–2:23 (2008)

[14] Pietro, R.D., Oligeri, G.: Silence is golden: Exploiting jamming and radio silence to communicate. ACM Transactions on Information and System Security (TISSEC) **17**(3), 9:1–9:24 (2015)

[15] Song, Y., Yang, C., Gu, G.: Who is peeping at your passwords at Starbucks?; To catch an evil twin access point. In: Dependable Systems and Networks (DSN), 2010 IEEE/IFIP International Conference on, pp. 323 –332 (2010)

[16] Sriram, V., Sahoo, G., Agrawal, K.: Detecting and eliminating Rogue Access Points in IEEE-802.11 WLAN - A multi-agent sourcing Methodology. In: Advance Computing Conference (IACC), 2010 IEEE 2nd International, pp. 256 –260 (2010)