On Detecting CTS Duration Attacks Using K-means Clustering in WLANs

Vishal Rajyaguru*, Bheemarjuna R Tamma†, B. S. Manoj††, and Mahasweta Sarkar*
* San Diego State University, San Diego CA 92182, USA
† Indian Institute of Technology Hyderabad, Hyderabad, India
††Indian Institute of Space science and Technology (IIST), Trivandrum, India

***Abstract*— IEEE 802.11 based Wireless LAN (WLAN) standard has been one of the most successful wireless technology standards with total expected sales rising to a staggering $6.1Billion by 2015. The proliferation of 802.11 based WLANs highlights the need to focus on development of new solutions for security as enterprises and campuses increasingly being covered by WLANs. Denial of Service (DoS) is one of the popular attacks that prevents WLAN users from accessing the wireless network resources. Most DoS attacks such as the Clear-to-Send (CTS) duration attack is easy to carry out by an attacker. This work focuses on the use of clustering techniques on wireless traffic datasets for detecting CTS-based DoS attacks on 802.11 WLANs. Performance evaluation shows that, under the cases of naïve CTS duration attacker as well as the sophisticated CTS duration attacker, the *k*-means clustering technique is able to achieve high detection rates and low false positive rates with relatively small values of *k (i.e., number of clusters)*.**

***Index Terms*— 802.11 WLAN, clustering, CTS duration attack, Denial of Service, K-means.**

I. INTRODUCTION

IEEE 802.11-based technologies continue to experience high growth rate. They come pre-installed in many consumer products like smartphones, tablets, cameras, printers, gaming consoles and laptop computers. The major issue with the 802.11 standard is the easiness with which an attacker can impact the network performance. Current wireless network standard has been designed under the assumption that nodes are honest and follow the protocol rules truthfully. An attacker can very easily and with low energy budget cause severe impact to the performance and availability of the wireless communication networks [1]. The IEEE 802.11 standard does provide limited support for confidentiality through the Wired Equivalent Privacy (WEP) or more advanced mechanisms. However, it has been shown to contain significant flaws by design [2]-[5]. Strengthening the security of IEEE 802.11 networks has been the focus of research activities for a long time. The 802.11 Working Group has come up with an enhanced version of the 802.11, known as 802.11i, to provide security mechanisms for wireless networks but there are security loopholes in 802.11i as well, as it can protect data frames only and an attacker can still spoof management or control frames to inflict significant damages [6]. Due to the inherent nature of the Radio Frequency (RF) medium and the design of the 802.11 protocols, wireless networks are very susceptible to Denial of Service (DoS) attacks.

DoS attacks at the physical layer aim at monopolizing the wireless medium (channel) by a continuous emission of radio signals effectively "jamming" the channel or by causing collisions to ongoing communication sessions resulting in retransmissions and wasted bandwidth. In addition to physical layer DoS attacks, an attacker can also launch attacks at the Medium Access Control (MAC) layer of 802.11-based WLANs. The most basic DoS attack at MAC layer involves spoofing the *Disassociation* and *Deauthentication* management frames. This type of attack is the most well-known and frequently launched DoS attack on WLANs. This attack floods the wireless medium with spoofed and modified authentication frames which will cause a client to become disassociated and generally causes erratic behavior. A more serious attack is the one that targets virtual carrier-sense mechanism in 802.11-based WLANs. This mechanism is used to reserve the channel for communication and mitigate collisions from hidden terminals. Each IEEE 802.11 frame carries a Duration field that indicates the number of microseconds that the channel is reserved. This value, in turn, is used to program the Network Allocation Vector (NAV) on each node. Only when a node's NAV reaches zero is it allowed to contend for the channel for transmission. This feature is used by the Request-to-Send (RTS)/Clear-to-Send (CTS) control frames in order to synchronize access to the channel. A significant advantage of MAC layer jamming over physical radio jamming is that the adversary node consumes less power in carrying out these attacks.

In this work, we report the development of a system that can classify CTS-based Duration attacks while still remaining efficient and protocol complaint. A wireless Intrusion Detection System (IDS), like the standard IDS, requires human resources to analyze and respond to threats. Such an IDS system should be designed keeping this in mind and also provide graphical visualization of alerts. Our system provides an early example for such visual IDSs.

Organization of the rest of the paper is as follows: Section II briefs the related work in this area, Section III presents the importance of the CTS attack detection, and Section IV explains our use of clustering for detecting the CTS duration attacks where naïve as well as sophisticated attack scenarios are considered. Section IV presents the results and finally Section V concludes the paper.

978-1-4673-5132-4/12/$31.00 ©2012 IEEE

II. Related Work

In response to security weaknesses of WEP protocol, the Wi-Fi Alliance created the Wi-Fi Protected Access (WPA) protocol. WPA is an encryption algorithm that attempts to take care of a lot of the vulnerabilities in WEP. It provides a way of ensuring the message integrity using MICHAEL and confidentiality using TKIP [20]. WPA is not an official IEEE standard, but is based on the IEEE 802.11i security standard, sometimes referred to as WPA2. While WPA does make cracking the shared key unfeasible, it stills suffers from availability attacks. Moreover, an attacker can launch more advanced attacks after successfully mounting a DoS attack [7]. DoS attacks are also known as availability attacks with the goal to disrupt the network service. There are surprisingly many ways to implement a DoS attack but they all have one of the two mechanisms in common: overwhelming the system by using up all its resources or executing an attack that exploits vulnerability in the communication media. Interfering with the wireless transmission, in order to carry out a DoS attack, is very easy to do. An attacker only needs a stronger transmitter and a strategic location to create availability problems. Moreover, as mentioned earlier, management packets in 802.11 are not authenticated, hence even a weak transmitter can keep legitimate users off the network.

A class of DoS attacks that affects the virtual carrier sensing mechanism works by exploiting the NAV field vulnerability. An attacker can manipulate the NAV value of wireless clients and, thereby, prevent them from accessing the channel for the duration of the NAV value. This can be achieved by sending a large number of frames with high duration field value [1]. While this is an effective way, a clever attacker would not use RTS or Data frames as they contain a Transmitter Address (TA) field. The TA is the address of a station that is physically transmitting the frame. A misbehaving node can be found if the number of RTS frames for that node exceeds the average number of RTS frames for the remaining nodes in the network [8]. Even if attacker is changing the MAC address of network interface card after every transmission, we can still deduce the identity of the device by using Radio Frequency Fingerprinting [9]. However, if the attacker is exploiting CTS packets, it becomes quite difficult to detect misbehavior since CTS packets do not carry TA. Furthermore, the fact that a node transmits elevated number of CTS packets does not imply a DoS attack is underway. An attacker capable of launching this attack would bring the network down on its knees and it will be even more difficult to detect this attack if the attacker is randomly changing the NAV duration, instead of keeping it at a constant high value. To detect this attack scenario, the administrator must examine every packet individually and see if it agrees with the enforced security policy for the network. Clearly, this task is not feasible for any network security administrator to perform because there are millions of packets transmitted through the network each day. If we can scale this problem down to a manageable size, it would aid in timely detection and elimination. Hence it can be identified as a problem of mining relevant information from a large collection of data. Thus, the problem of intrusion detection can be reduced to a data mining task of classifying data (network events) into difference classes (normal activity, attacks) as accurately as possible.

Using data mining for network security analysis is not a new idea [10]-[12]. As a matter of fact, data mining has been used to obtain training data for anomaly based intrusion detection systems. Anomaly-based IDSs sound an alert if the network behavior deviates from the expected behavior of the network. Anomaly-based IDSs have the ability to detect novel attacks for which signatures may not yet available. However, in practice, it is difficult to obtain an accurate and comprehensive profile of the network behavior. To overcome this weakness, research has been conducted to train anomaly systems using unlabeled data [13], [14].

Clustering is a valuable machine learning tool in data mining. Clustering is based on the principle of maximizing the intraclass similarity and minimizing the interclass similarity. The hope in using clustering for IDSs is that intrusive elements will cluster together whereas normal elements will cluster separately. Moreover, cluster centers are representatives for each cluster. Therefore, examining the cluster center will quickly enable us to see if cluster members are a part of normal component or if they need further examination.

III. Why CTS attack Detection Is Challenging?

CTS attacks can be more severe than RTS attacks because it is easy to detect and prevent an RTS attack using NAV timer delays [1]. Furthermore, CTS frames are very small and simple yet the potential damage that can be caused is quite high. Further, all protocol compliant nodes cannot ignore CTS frames and the duration field contained in them.

Given that a significant number of existing wireless devices have 802.11b adapters, many WLANs will be forced to deal with B/G coexistence for at least another decade. The IEEE 802.11 standard mentions protection mechanism using either RTS/CTS or CTS-to-Self. Most devices will use CTS-to-Self mechanism as it adds lower overhead and better throughput compared to RTS/CTS. A CTS-to-Self packet is an ordinary CTS packet that carries the transmitter's own MAC address. An attacker informed with this knowledge can misuse the CTS-to-Self mechanism to deny network availability. Furthermore, since the mechanism only requires the transmitter's own address, an attacker can easily misguide the protection system by using MAC address spoofing.

These advantages make CTS-based attacks very attractive for an attacker as it is very effective, simple to implement and hard to detect.

IV. Clustering for CTS attack detection

The aim of data mining is to make sense of large amounts of mostly unsupervised data, in some domain [16]. Clustering is a type of data mining model which analyzes data objects without consulting a known class labels i.e., it is a type of

unsupervised learning technique. Clustering techniques can be applied to data that is quantitative, qualitative, or a mixture of both. In this paper, the clustering of quantitative data is considered. Various definitions of a cluster can be formulated, depending on the objective of clustering. Generally, a cluster is a group of objects that are more similar to one another than to members of other clusters. The term "similarity" (or distance) should be understood as mathematical similarity, measured in some well-defined sense. The distance $d(\mathbf{x}, \mathbf{y})$ between $\mathbf{x}$ and $\mathbf{y}$ is considered to be a two-argument function satisfying the following conditions:

$d(\mathbf{x}, \mathbf{y}) \geq 0$ for every $\mathbf{x}$ and $\mathbf{y}$
$d(\mathbf{x}, \mathbf{x}) = 0$ for every $\mathbf{x}$
$d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$

In our case, we have chosen the distance function to be the Euclidean distance. Given a database of n objects, the partitioning algorithm constructs k partitions satisfying the following requirements:

- Each cluster must contain at least one object and
- Each object must belong to exactly one cluster.

Given the number of cluster partitions to create, k, the algorithm creates an initial partitioning. It then uses an iterative relocation technique that attempts to improve the partitioning by moving objects from one group to another. The most well-known and commonly used partitioning methods are k-means and k-medoids. K-means algorithm is more suitable for analysis of network traffic pattern as it is very sensitive to outliers (anomalies) and it has very low time complexity $O(knl)$, where l is the number of iterations that the algorithm is performing [17]. A low time complexity is valuable for IDS so that appropriate action can be taken in a timely fashion.

For clustering to produce meaningful results, proper choice of similarity parameters must be made. Since this type of attack is carried out by manipulating the Duration field, we will use it as our first dimensional axis. We have chosen Received Signal Strength Indicator (RSSI) as our second dimension. It is a mechanism by which RF energy is to be measured by the circuitry on a wireless antenna. Higher value of RSSI indicates a stronger signal strength implying closer proximity of the antenna and vice versa. RSSI is a robust measurement that is hard to forge arbitrarily and it is correlated to the antenna's location [18]. As an added advantage, once attack packets are discovered, RSSI measurement can be used to make an estimation of the attacker's physical location as well as RF fingerprint [19].

A. Our Approach

We have created a testbed for studying the effectiveness of clustering techniques for identifying CTS-duration attacks. We deployed a large number of wireless network traffic monitors that gather network traffic traces from multiple Wi-Fi channels from the UC San Diego campus wireless network and stored them in a central database repository. We modified some packets to simulate attackers where the Duration field is dynamically or statically altered. A separate detection utility read the packet traces stored in the database and feed them to the k-means clustering technique for detecting CTS-Duration attack incidents. Our database contains header information for physical layer and data link layer, along with the higher-level layer header fields. The attacker can control frequency of the attack packets as well as the duration for those packets. Before using this system for detecting anomalies, the system administrator must have a security policy in place and must be aware of the network behavior under normal circumstances such as data rate, traffic pattern, congestion periods, etc. This information is needed to distinguish between what is normal and anomalous. Additionally, knowing what attack scenario to look for is helpful since different attack scenarios have differing value of k that is optimal. Conversely, given an unknown attack scenario, the administrator can calculate the value of k and match it with a list of known attack cases. In this way it can be used to discover and categorize novel attacks.

In this work we have considered two attack scenarios: (i) an attacker who sends CTS packets with high (but constant) duration values in order to monopolize the channel which we classify as a case of Naïve CTS duration attack and (ii) an attacker who is able to vary the duration value randomly so as not to be detected easily. The second case is considered as a case of sophisticated CTS duration attack. An attack scenario was enacted by introducing anomalous packets into the database. For the first case, the duration was set to a high value of 30 milliseconds. Thus, an attacker needs to only transmit roughly 35 packets per second to disrupt the channel. For the second case, the duration was set to a random value instead. Before applying clustering to the dataset, pre-processing was done to remove MAC packets with power bit set as in such case the duration value is station ID.

We now present the results we obtained and our interpretation for these two cases. In the following plots, x-axis represents RSSI signal strength and y-axis represents duration value of CTS packets.

B. Detecting Naïve CTS duration attack.

The Attacker transmitted 144 packets which represented 1.3% of the total traffic. The total number of CTS packets in the dataset was 10,768. The remaining points represent normal data. We ran the test three times with varying the RSSI value of the attacker each time in order to simulate the effects of an attack by varying attacker's signal strength.

Figure 1 shows the input dataset before clustering was applied. For the sake of clearer representation, the values obtained from the dataset have been normalized to the range of 0 to 1. Attacker's packets are located at (1, 1) on the plot in Figure 1. The value of attacker RSSI signal strength was 75.

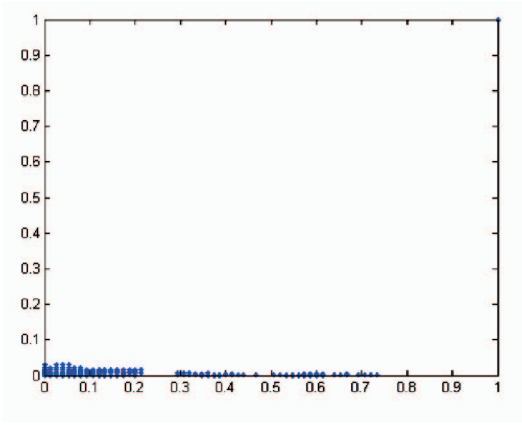

Fig. 1. Input dataset for naïve attack where RSSI value is 75. (X-axis represents normalized RSSI values and Y-axis normalized CTS-Duration values)

We ran the *k*-means clustering algorithm iteratively with increasing value of *k* and were able to fully classify the attack with $k = 5$. Furthermore, increasing the value of *k* beyond 5 did not affect the performance or accuracy of the system except adding additional overhead needed for the expert to examine more clusters. Figure 2 shows the result of clustering with 5 clusters. Each cluster is represented by a unique color for easy differentiation and cluster centroids are represented by a red circle. Since the cluster center is the representative for the corresponding cluster, examining it will quickly enable to see if cluster members are part of normal behavior or if they need further examination.

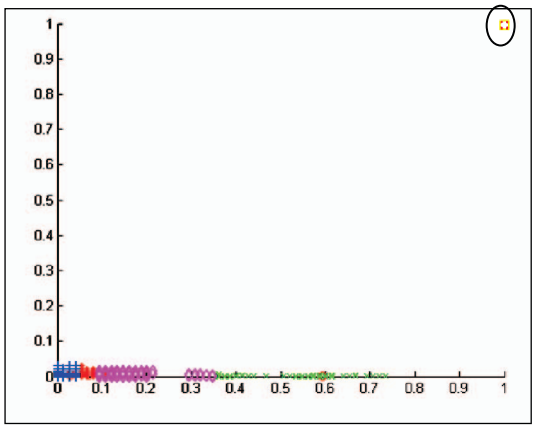

Fig. 2. Clustering result for naïve attack with $k = 5$ and RSSI value of 75 (X-axis represents normalized RSSI and Y-axis represents normalized CTS-Duration values).

C. *Detecting Sophisticated CTS duration attack.*

In this case we assume that the attacker is clever enough to randomly change the contents of the Duration field. By doing so, an attacker's average duration value becomes closer to the average for the rest of the network. Such an attack would be very difficult to detect using traditional methods.

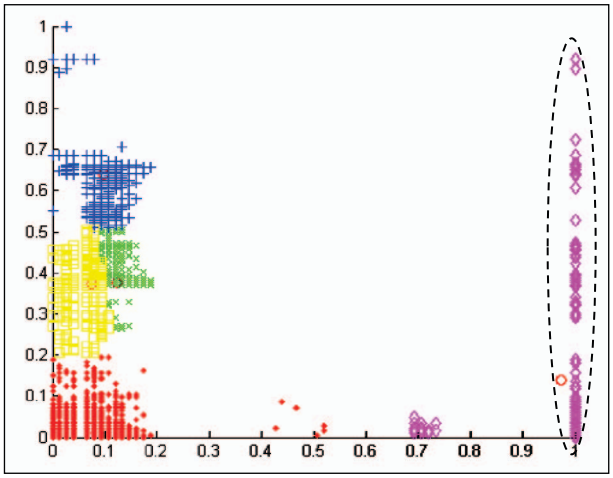

Fig. 3. Clustering result for sophisticated CTS attack with $k = 5$ and RSSI value of 75 (X-axis is normalized RSSI and Y-axis is normalized CTS-Duration values).

Here the attacker transmitted 700 packets which represented 7.2% of the total network traffic. We again ran this test for three different RSSI values to see if we reach the same conclusion. First, as in previous case, we ran our algorithm with increasing value of k and found that $k = 10$ yielded the best solution for any case in this attack scenario. Figure 3 shows the result of clustering with $k = 5$. Attack packets are encircled with a dotted line in the plot. Figure 4 shows the result with $k = 10$ and we are able to completely classify the attack in a single cluster.

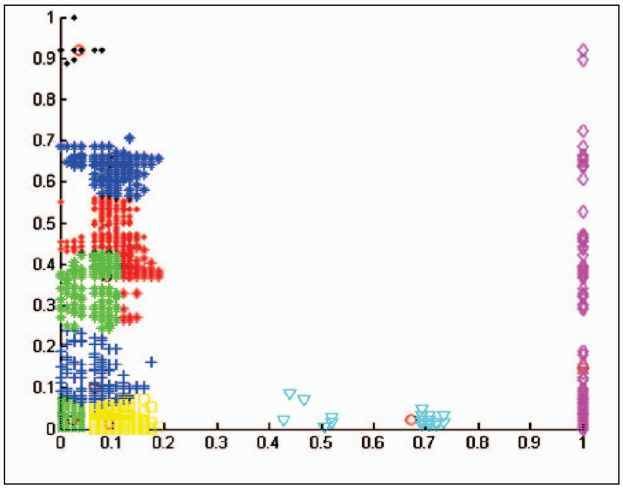

Fig. 4. Clustering result for sophisticated CTS attack with $k = 10$ and RSSI value of 75. (X-axis is normalized RSSI and Y-axis is normalized CTS-Duration values). The attack cluster is the right most one with purple color.

The second test was performed with a lower RSSI of 31. The attacker again transmitted 700 packets which are located at x = 0.56. Figures 5--7 show the clustering result with three different values of *k*. We yet again achieve the best performance at $k = 10$.

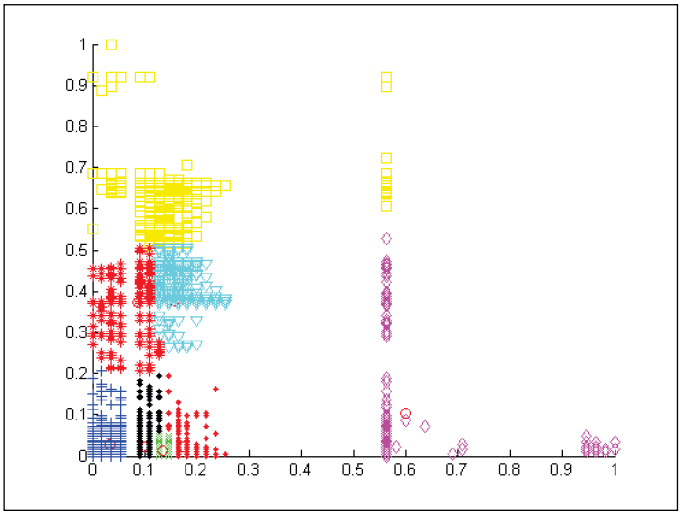

Fig. 5. Clustering result for sophisticated attack with k = 8 and RSSI value of 31. (X-axis is normalized RSSI and Y-axis is normalized CTS-Duration values). The attack cluster is the purple colored one with diamond shaped markers.

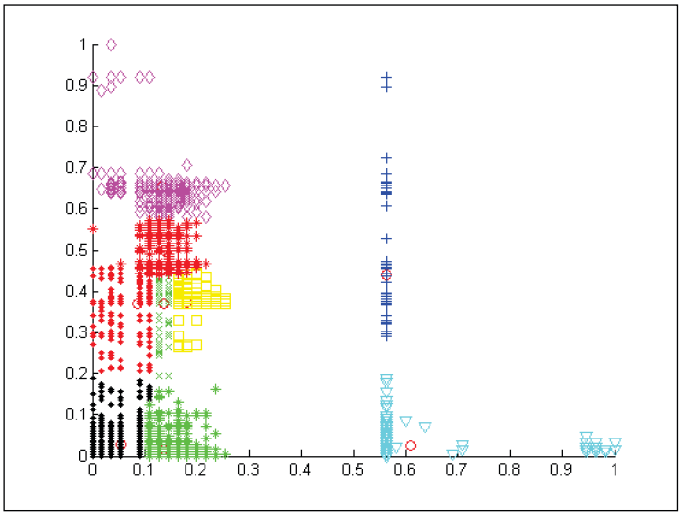

Fig. 6. Clustering result for sophisticated attack with k = 9 and RSSI value of 31. (X-axis is normalized RSSI and Y-axis is normalized CTS-Duration values). Attack clusters are identified as Blue plus and Light Blue triangle markers.

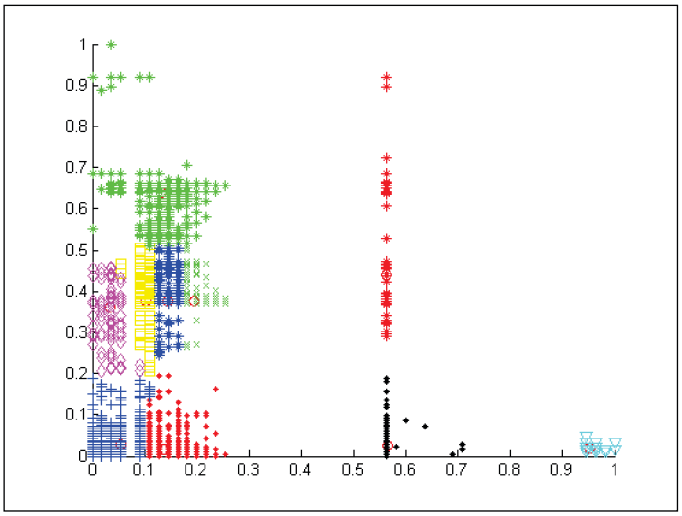

Fig. 7. Clustering result for sophisticated attack with k = 10 and RSSI value of 31. (X-axis is normalized RSSI and Y-axis is normalized CTS-Duration values). Two attack clusters are Red stars and Black diamonds.

V. Results

The goal of any system is to minimize inaccuracy while also maintaining performance. The following metrics are used to judge the efficacy of proposed system.

- *False Negative Rate*: False negatives are the number of anomalous instances incorrectly classified as part of normal traffic. If attack packets are grouped with normal packets, there is a high likelihood that they would go undetected. False negative rate is calculated across all the clusters that are classified to be normal. The goal here is to minimize the false negative rate.
- *False Positive Rate*: False positives are the number of instances of normal traffic incorrectly classified as anomalous. False positive rate is calculated using the cluster size of anomalous cluster(s). The goal is to minimize the false positive rate.
- Detection Rate: Detection is the number of anomalous packets that the administrator detects.

In the case of naïve CTS duration attack, when the attacker is maintaining a constant duration, we were able to achieve a 100% detection rate with no false positives implying that k-means clustering is a great tool for detecting this type of attack. Furthermore, we achieved desired accuracy at the same value of k irrespective of the received signal strength of packets injected by the attacker. For our test dataset, the value of k was 5 which have been verified by calculating the test metrics. Figure 8 shows the analysis. We ran the same attack scenario at RSSI value of 31 and again at RSSI value of 16 and were able to obtain similar results at the same value of k.

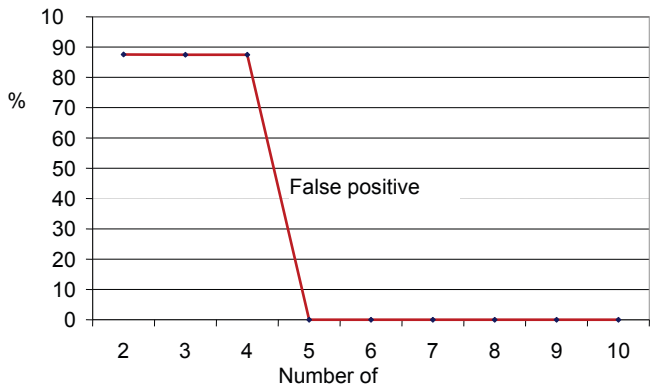

Fig. 8. Result Analysis after applying k-means clustering to a naïve attack.

From the results, we conclude that clustering is able to classify CTS-duration attack with a very low value of k even when the attack packets represent a very small number of total network traffic.

The second case that we considered is even more difficult to detect using traditional methods. The attacker is shrewdly introducing variations in the CTS-duration value such that the average duration when calculated will be close to normal

network behavior. Similar to the previous case we are able to achieve 100% detection rate as shown in Figure 9.

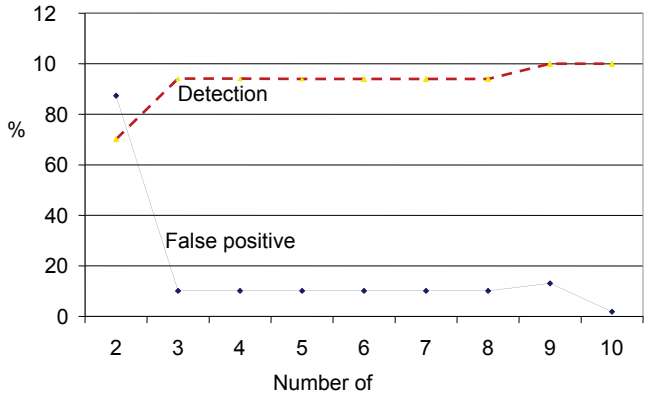

Fig. 9. Result analysis after applying k-means clustering to sophisticated CTS attack with RSSI value of 31.

One thing worth noting here is the slight peak in false positive rate near the end. This is due to the peculiarity with the definition of false positives. When the number of clusters is high, subtle variation in cluster centers can change membership function, i.e., the cluster size for that cluster. The definition of false positive takes into consideration the cluster size hence it is very sensitive to variation in membership function. Table I lists the results we obtained for our test run. In Table I, the value in a cell is the cluster size and cells that are colored contain anomalous packets.

TABLE I
RESULT ANALYSIS OF SOPHISTICATED ATTACK.

K	2	3	4	5	6	7	8	9	10
C1	5825	5654	4692	2706	972	2253	422	1098	1755
C2	3882	3320	733	1990	2904	847	770	1727	953
C3		733	974	3307	2297	3304	1002	209	1552
C4			3308	972	1012	732	974	1693	889
C5				732	1790	567	733	806	251
C6					732	1694	3544	565	65
C7						310	1114	1553	500
C8							1148	305	209
C9								1751	933
C10									2600

VI. CONCLUSION

In this work we proposed and implemented a method to identify CTS-duration based DoS attacks in WLANs. We have considered two cases: (i) the naïve attacker case where the attacker is blocking access to the channel by sending CTS packets with high duration field value and (ii) a sophisticated attacker case where the attacker is dynamically changing the duration value. Our implementation of clustering was able to identify both attacks with high degree of accuracy. Under the cases of Naïve CTS duration attack as well as the sophisticated CTS duration attack, k-means clustering was able to achieve high detection rates with relatively small values of *k* (between 5 and 10).

REFERENCES

[1] J. Bellardo and S. Savage, "802.11 Denial-of-Service Attacks: Real Vulnerabilities and Practical Solutions" Proceedings of the 12th conference on USENIX Security Symposium, 2003.

[2] N. Borisov, I. Goldberg, and D. Wagner, "Intercepting Mobile Communications: The Insecurity of 802.11." 7th Annual Int'l. Conf. Mobile Comp. and Net., Rome, Italy, 2001.

[3] J. Walker, "Unsafe at Any Key Size: An Analysis of the WEP Encapsulation," IEEE 802.11 Task Group E, 2000.

[4] S. Fluhrer, A. Shamir, and I. Mantin, "Weaknesses in the Key Scheduling Algorithm of RC4," Sel. Areas of Cryptography, Toronto, Canada, 2001.

[5] W. A. Arbaugh, "An Inductive Chosen Plaintext Attack Against WEP and WEP2. 2001," IEEE 802.11 Working Group, Task Group I (Security), 2002.

[6] He C., Mitchell J., "Security Analysis and Improvements for IEEE 802.11i", 12th Annual Network and Distributed System Security Symposium (NDSS'05), 2005.

[7] M. Lynn and R. Baird, "Advanced 802.11 attack", Black Hat Briefings, July, 2002.

[8] Thamilarasu G., Mishra S., Sridhar R., "A Cross-layer Approach to Detect Jamming Attacks in Wireless Ad hoc Networks", IEEE Military Communications Conference, 2006.

[9] Hall J., "Detection of rogue devices in Wireless Networks", Ottawa-Carleton Institute for Computer Science, PhD Thesis, 2006

[10] Barbara D., Jajodia, S., "Applications of Data Mining in Computer Security", Kluwer Academic Publishers, 2002.

[11] Levent E. et al, "MINDS – Minnesota Intrusion Detection System", Data Mining – Next Generation Challenges and Future Directions, MIT Press, 2004.

[12] Wenke L., Salvatore S., Kui M., "A Data Mining Framework for Building Intrusion Detection Models", IEEE Symposium on Security and Privacy, 1999.

[13] Khoshgoftaar, T.M., Seiffert, C., Seliya, N., "Labeling Network Event Records for Intrusion Detection in a Wireless LAN", IEEE International Conference on Information Reuse and Integration, 2006.

[14] Portnoy, L., Eskin E., Stolfo S.J., "Intrusion Detection with unlabeled data using clustering", Proceedings of ACM Workshop on Data Mining Applied to Security, 2001.

[15] IEEE 802.11g, "Further Higher Data Rate Extension in the 2.4 GHz Band", MAN Standards Committee of the IEEE Computer Society, 2003.

[16] Cios. K., et al, "Data Mining. A Knowledge Discovery Approach", Springer, 2007.

[17] Simovici D., Djeraba C., "Mathematical Tools for Data Mining", Springer, 2008.

[18] Yong S., Keren T., Guanling C., David K., Andrew C., "Detecting 802.11 MAC Layer Spoofing Using Received Signal Strength", IEEE Infocom 2008.

[19] Lau E., et al, "Enhanced RSSI-Based Real-Time User Location Tracking System for Indoor and Outdoor Environments", IEEE International Conference on Convergence Information Technology, 2007.

[20] IEEE 802.11, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications", Standard Committee, IEEE, Aug. 2007.

[21] Rajyaguru V., "Proximity based WLAN Intrusion Detection System", San Diego State University, Department of Computer Science, Masters Thesis, 2010.