Honeypot based Cellular Cross-layer Intrusion Detection and Response

Md. Motaharul Islam¹ Ali Alzahrani¹ Mohammad Raihan Kabir² Rifat Rahman³
¹Dept. of Computer Science, Faculty of Computer and Information Systems
Islamic University of Madinah, Madinah, Kingdom of Saudi Arabia
²ICT Division, Al-Arafah Islami Bank Ltd. Bangladesh
³Dept. of Computer Science, TU Clausthal, Germany

Abstract
Data produced by cellular devices and its usage are growing exponentially due to the popularity of smart phones and tablets. Because of their smaller sizes and reduced capabilities, it is sometimes hard to impose adequate security measures to these everyday used devices. For this reason, they become a popular target for data theft and misuse. Due to the nature of cellular network various kinds of attack on the cellular devices such as distributed denial-of-service (DDoS) are happening frequently. Conventional security measures for the layered architecture cannot cope with this ever-evolving security issue. Thus, we have proposed an enhancement of cross-layer design for security in wireless network entitled as Cellular Cross-layer Intrusion Detection and Response (CXMIDR) by using honey pot approach. We have utilized the cross-layer design which integrates features from various layers for detecting intrusions in wireless environment and our proposed system enhances performance up to 20%.

Key-words:

1. Introduction

Cellular network is a special type of wireless network distributed over large areas called cells, through the use of radio towers which are commonly known as base stations. These base stations work as a transceiver. In the network, each cell uses different frequencies from their neighbors, to avoid interference and provide bandwidth for necessary data transmission and through the use of smart phones and tablets a large amount of data is being transmitted every day where wireless technology is used as a backbone. A wireless network is any type of computer network that uses wireless data connections for connecting network nodes. Wireless networks are generally implemented and monitored using radio communication. Though wireless network has more advantage compared to wired network but still security of wireless network is a great issue. Some reasons of wireless network’s being unsecured are mentioned in [1] and [5]. The risks to users of wireless technology have increased as the service has become more popular. There were relatively few dangers when wireless technology was first introduced. However, there are a great number of security risks associated with the current wireless protocols. Different kinds of security attacks are mentioned and discussed in details in [1] and [2]. Some of the renowned attacks are DOS attack, Sybil attack, Wormhole attack etc. Security solutions may be authentication, cryptography or key management. Security solutions are available recently nevertheless, they alone can’t prevent any kind of wide range of attacks. Normally, security measures are provided in the application layer. Layered approach is fit for wired network but not for wireless network. In [3] the solution of this is discussed. The researchers are now interested in cross-layer design techniques to alleviate the problems of layered architecture. Some reasons of choosing cross-layer design are mentioned in [5].

Cross-layer design which is unlike the layered architecture refers to designing of interfaces, protocols or architectures that utilizes inter-layer interactions that is a superset of the standard interfaces for achieving better performances and facilitating new features. One of the most important features of cross-layer design is sharing information between two or more layers. The idea behind cross layer information exchange is to use various parameters from different layers for joint optimization of protocols across the communication stack. In [3] different cross-layer design approaches are discussed briefly. Cross-layer architecture and its security as an emerging concept have many opportunities. There are many proposals regarding this topic. Physical layer data can be passed to application layer and reverse is also possible. Besides, adjacent two or three layers can be merged together that is called a super-layer. Research scope regarding security in cross-layer is increasing recently and many works have been done. But yet, implementation is not that much satisfactory.

Intrusion detection system (IDS) is an important security aspect in wireless network. It detects the misbehaving and malicious nodes and isolates them. Researchers have recently proposed some IDS approaches which is discussed in [1]. From [1] and [4] it is noticed that IDS can classify attacks by misuse, anomaly or specification-based
The main contributions of this paper are as follows:
1. We have proposed a cross layer Intrusion Detection System based on honey pot approach that combines both wireless network and cross layer framework which is better than its counterparts in many aspects.
2. In case of false positive and false negative reduction purpose, our system works 3%-5% and 5%-8% better than its counterparts respectively.
3. We compare between CXIDR, XIDR and single layer, our system performs the best among the three.
4. We have also introduced various packet management techniques to supply our detection techniques information with least amount of time.
5. We have also proposed two databases which will store all the information about the attacks and our responses in respect to them. Just by detecting we cannot solve our problem. We have to response to the threats. So, we have also proposed a response module engine which will select a cost-effective response.

The rest of the paper is organized as follows. Section 2 reviews background. Section 3 describes the proposed framework in details. Section 4 shows the performance evaluation, experimental and simulation results. Section 5 finally concludes the paper.

2. Background

2.1 Intrusion Detection

Intrusions are the actions that break the security policy of the system. Intrusion detection is the process which is employed to detect such intrusions. In cellular network, intrusion detection techniques can be categorized into two methodologies: anomaly detection which is based on the normal behavior of a subject or misuse detection which is based on attributes of known attacks or system vulnerabilities [4].

2.2 Classification of ID Techniques

There are two main categories: statistical and rule-based. And the other categorization is between host based and network based intrusion detection. The statistical approach employs other statistical methods to obtain metrics that model the behavior of a user [11]. Anomalies detection is not on the basis of predefined rules and so it has the positive features of detecting new attacks. The drawbacks of statistical detection are that, it requires huge amount of statistical data to build the threshold/profiles. A rule based approach [12] has a set of rules that define normal behavior. A rule based anomalies system is more or less similar to statistical anomaly detection. The rules are set up from the previous behavior pattern. Rule based anomalies face lots of challenges, namely, monitoring etc.

2.3 Related Works

‘Cross Layer Design’ has become a new buzz word in wireless technology. Through experience the researchers have learned that there are many problems regarding wireless network which do not occur in wired networks. To cope up with these new kinds of problems collaboration between various OSI layers have been proposed. In [3] Vineet Srivastava et al. have tried to give us a rough picture of cross layer architecture and various proposals given by researchers regarding this. The authors have given us a clear idea about how cross layer design can be implemented such as: creating new interface between layers, creating super-layers, designing coupling without creating new interfaces or by vertical collaboration. They also have described how different layers can pass information among each other. They proposed three methods like direct communication (through data packets), shared database or by creating new abstractions.

In [4] Jatinder Singh et al. proposed a new method to detect intrusion with the help of cross layer design. In the proposed mechanism, the Received Signal Strength (RSS)
and the TT value (the time it takes for RTS-CTS handshaking) are the identifying merit here. A dynamic profile is created based on these measures. As no two devices can have the same values it is a very good identifier. If a node has higher RSS value than the threshold then that device is identified as an attacker. The threshold values are updated periodically and are managed in the base stations.

Igors Svec et al. presented a complete intrusion detection and response framework named Cross-layer Intrusion Detection and Response (XIDR) in [6]. According to them this framework utilizes multi-source IDS and cross-layer automated intrusion response system to deploy cost-effective and efficient pre-emptive responses. It is used to detect intrusions in wired environment. They have provided a model which includes multiple intrusion detection sources, data sources, automated response selection engine and a collection of response deployment modules. The authors aimed to improve IDS by moving intrusion detection from user space to kernel space. They have also improved the response engine and provided a diagram of XIDR deployed in test-bed environment. Their proposed framework was Host Intrusion Detection System (HIDS) that runs on individual hosts in the network and it monitors the incoming and outgoing packets from the hosts and will alert the user or administrator when suspicious behavior is detected. Finally, the authors have shown some simulation results which shows the efficiency of cross-layer approach over the single layer approach in detecting intrusions.

2.4 Major challenges

In our endeavor to develop a new IDS system we have faced various fundamental challenges. Some of them we have overcome, some of them we still are trying to get our head around. The major challenges are briefly explained below:

i. **Wireless Nature of Communication:**
   As the network architecture is wireless, packets may come from anywhere in a broadcast fashion. That means anyone can pick up the packet and modify, change or simply fabricate it.

ii. **IP Address Spoofing:**
   IP addresses are dynamically allocated to mobile devices but it sometimes let the attacker impersonate as a legitimate user.

iii. **Implementing IDS in cross layer design:**
    In wireless architecture, implementing IDS itself is a challenge. To use cross layer design is probably one of the newest research points in this sector which is difficult too.

iv. **Creating NIDS:**
    All the research about a wireless IDS system has been host based. In this respect, NIDS system has been relatively untouched. So, we have to carefully consider various aspects of the network and the system’s feasibility against HIDS.

3. Our proposed framework

We are proposing a new cross layer IDS system which will effectively perform its task against any kind of intrusion. Our proposed architecture has 4 modules. These are as follows:

1. Intrusion Detection Module (IDM).
2. Detection Database.
4. Response Deployment module (DSRM).

In Fig. 1, detail architecture of our proposed system has been shown.

Fig. 1  CXIDR Framework.

1. Intrusion Detection Module (IDM):
   Our intrusion detection module combines various techniques and implements cross layer architecture for detection method. In Fig. 2 IDM has been shown in detail.

Fig. 2  Intrusion Detection Module
A. Packet Decoder

Our system’s detection techniques sometimes need the packet to be accessed in different layers. But in a base station all packet is relayed to its destination. So, it doesn’t need to decode the packets. So, we introduce our own packet decoder. With this we will decode our packet to detailed analysis through various techniques such as Snort, RSS or ModSecurity. These methods also need the headers attached by various layers of open system interconnection (OSI) model.

B. Pre-processor

After decoding now, we have the headers of several layers of OSI models like physical, network, transport, application layer etc. As our techniques use different layer headers and the data it encapsulates, we supply those modules with the proper headers and data segment. In this way we can save time which would have been wasted if they had to sort.

C. Detection Sources

We use various detection techniques. They give us the ability to analyze a packet in a cross-layer environment. These techniques are briefly discussed below:

I. Snort

Snort is an open source intrusion detection system. It combines the benefits of signature, protocol, and anomaly-based inspection. It is the most widely deployed IDS/IPS technology in modern world.

II. ModSecurity

Array of request filtering and other security features is supplied by ModSecurity to the Apache HTTP Server, IIS and NGINX. ModSecurity is a web application layer firewall. It is an open source software.

III. RSS

RSS means Received Signal Strength. This value is unique to the sender and receiver. When a call is initiated the signal strength from the sender and receiver is recorded and the time it takes to reach them is also recorded. When the packets pass through the base station than the RSS value is checked. If the RSS value is greater than we understand that there is an intruder in the network. So, we treat the packets with higher RSS value an intrusion and take necessary action. After the incoming packet has gone through all the sources, if an intrusion is detected then the necessary information is passed to Dynamic Response Selection Module through Detection Database and then an appropriate response is generated. A detail of RSS is described in [4]. RSS can be achieved by RSS Indicator (RSSI). RSSI is a function of the distance between two nodes and can be computed as follows:

\[ \text{RSSI}(d) = \text{RSSI}(d_0) - 10 \log(d/d_0) \]

In above equation, \( \text{RSSI}(d) \) is the receiving signal strength in decibel (db) at a distance \( d \) from the source node. \( \text{RSSI}(d_0) \) is the receiving signal strength at a distance \( d_0 \) from the source and \( n \) is the attenuation exponent [14].

IV. Honeypot

Honeypot is a resource which is intended to be attacked and facilitate to gain more information about the threat and threat techniques. It can be used to attract and divert the attacker from the real targets. It will sit idle listening and waiting for something of interest. Our honeypot machine is based on cross layer information from snort, mod-security and receive signal strength. By collecting information from different layers the machine fools the attackers into believing that it is the real system so that the administrator can effectively log information regarding the attacker's behaviors. In this way, the system administrator can learn about the vulnerabilities of the current system and redesign it to more secured.

2. Detection Database:

Detection Database stores all the information of our intrusions detected so far and all the intrusion that we have detected in the life time. There are two kinds of database:

i. Temporary Database:

This database stores all the information about the recent attacks like attack type, layers associated, threat level, generation source and all the info about the source generation it. After a certain amount of time the information is saved into the permanent database.

ii. Permanent Database:

This database stores all attacks the system has faced so far and the responses to them. It stores a detailed description from the attack occurring to the time where the system has deemed the attack to be nullified. It stores the information on which responses were used and how much cost effective it was. It is a permanent sink for all the information and the system’s effectiveness in it for future consultation.

![Figure 3. Detection Database.](image-url)
After receiving all the necessary information about the current attack and if there was a similar past attack from the same or different source a cost-effective response is generated. To do so we need two things. They are:

I. Response Selection Engine

II. Response Rules Database

Response Selection Algorithm:

Input: (ID, DATETIME, SOURCE_IP, DESTINATION_IP, DESCRIPTION, TYPE, SOURCE_MODULE, DEPLOYMENT_MODULE, PRIORITY, RELATED_LAYERS, RESPONSE).CXIDR_TEMPORARY_DATABASE

(RESPONSE_ID, DESCRIPTION, SOURCE_MODULE, DEPLOYMENT_MODULE, TYPE, RELATED_LAYERS, PRIORITY, COST, RESPONSE).CXIDR_RESPONSE_RULES

Output: RESPONSE

Steps:
Select all RESPONSE where

\[
\text{PRIORITY.CXIDR\_TEMPORARY\_DATABASE} == \\
\text{PRIORITY.CXIDR\_RESPONSE\_RULES} \\
\quad \&\& \quad \\
\text{RELATED\_LAYERS.CXIDR\_TEMPORARY\_DATABASE} == \\
\text{RELATED\_LAYERS.CXIDR\_RESPONSE\_RULES} \\
\]

If (Count(RESPONSE)>1){
Select RESPONSE with Min (COST);
}

Else If (Count(RESPONSE)==1){
Select RESPONSE;
}

Else{
Select DEFAULT RESPONSE;
}

In summary, using the response selection algorithm we select the response by matching the priority and OSI layer from the two tables CXIDR_TEMPORARY_DATABASE and CXIDR_RESPONSE_RULES. If multiple responses are found, our response selection engine will select the response from the CXIDR_RESPONSE_RULES with the lowest cost. If there is only a single response is found, that response will be selected by the selection engine. But if there is mismatch in the two tables, no response will be selected and a default response will be generated.

II. Response Rules Database:

This database stores all the responses and it can be easily updated. Whenever the Response Selection Engine needs a response, they are generated from here

4. Response Deployment Module:

This module deploys the responses selected by DRSM. At first, the response is received from the DRSM and it follows necessary protocol to carry out the response. It also will calculate success rate and then send that information to Detection Database for future storage and consultation.

4. Performance evaluation

We have used Network Simulator-3 or NS-3 to simulate RSS. We have also compared our system’s performance with Single Layer and XIDR. In Fig. 5 and Fig. 6 false positive and false negative reduction is shown.
In both figure, packet rate is in X-axis. In Fig. 5 false positive rate and in Fig. 6 false negative rate is in Y-axis. So, from both this figure it has been shown that our system performs better than others.

Table 1: False Positive Reduction Comparison

<table>
<thead>
<tr>
<th>Packet History</th>
<th>Single Layer</th>
<th>XIDR</th>
<th>CXIDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0.5</td>
<td>0.4</td>
<td>0.2</td>
</tr>
<tr>
<td>4</td>
<td>1.3</td>
<td>0.8</td>
<td>0.6</td>
</tr>
<tr>
<td>8</td>
<td>2.3</td>
<td>1.5</td>
<td>1.5</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>4</td>
<td>3</td>
</tr>
</tbody>
</table>

In Table 1 and Table 2 there are two table of comparisons showing the false positive and false negative comparison respectively among Single Layer, XIDR and CXIDR. This comparison is done with respect to random packet rate. These data are used to draw the graphs of Fig.5 and 6. The graphs are drawn with the help of MATLAB. We have collected data manually from Snort, RSS and ModSecurity to draw the graphs used for performance analysis.

In Fig. 7 intrusion detection comparison among single layer, XIDR and CXIDR is shown and intrusion detection rate is in Y-axis. From this figure it is shown that CXIDR has more success in detecting intrusions than other systems.

We have used Oracle Database 10g Express Edition to create various tables which is used in our algorithm. To implement our algorithm, we connected the Oracle with Java using NetBeans IDE 7.1.2. For our implementation purpose we have created two tables in Oracle which are shown in Table 3 and Table 4.

Table 2: False Negative Reduction Comparison

<table>
<thead>
<tr>
<th>Packet History</th>
<th>Single Layer</th>
<th>XIDR</th>
<th>CXIDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0.4</td>
<td>0.35</td>
<td>0.2</td>
</tr>
<tr>
<td>4</td>
<td>1.2</td>
<td>0.75</td>
<td>0.5</td>
</tr>
<tr>
<td>8</td>
<td>2.2</td>
<td>1.8</td>
<td>1.5</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>4</td>
<td>2.8</td>
</tr>
</tbody>
</table>

Table 3: CXIDR_TEMPORARY_DATABASE

<table>
<thead>
<tr>
<th>ID</th>
<th>DATETIME</th>
<th>SOURCE_IP</th>
<th>DESTINATION_IP</th>
<th>DESCRIPTION</th>
<th>TYPE</th>
<th>SOURCE_MOD</th>
<th>DEPLOYMENT_MOD</th>
<th>PRIORITY</th>
<th>RELATED LAYER</th>
<th>COST</th>
<th>RESPONSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>01A</td>
<td>11/07/2014</td>
<td>10.20.62.31</td>
<td>10.20.62.31</td>
<td>XXX</td>
<td>YYY</td>
<td>SHORT</td>
<td>SRC</td>
<td>100</td>
<td>RSS</td>
<td>2</td>
<td>CXIDR</td>
</tr>
<tr>
<td>02A</td>
<td>13/07/2014</td>
<td>10.20.62.35</td>
<td>10.20.62.35</td>
<td>PGP</td>
<td>RSS</td>
<td>SRC</td>
<td>SRC</td>
<td>100</td>
<td>RSS</td>
<td>2</td>
<td>CXIDR</td>
</tr>
<tr>
<td>03A</td>
<td>21/07/2014</td>
<td>10.20.62.31</td>
<td>10.20.62.31</td>
<td>PGP</td>
<td>RSS</td>
<td>SRC</td>
<td>SRC</td>
<td>100</td>
<td>RSS</td>
<td>2</td>
<td>CXIDR</td>
</tr>
<tr>
<td>04A</td>
<td>28/07/2014</td>
<td>10.20.62.34</td>
<td>10.20.62.34</td>
<td>LIGHT</td>
<td>SRC</td>
<td>SRC</td>
<td>SRC</td>
<td>100</td>
<td>RSS</td>
<td>2</td>
<td>CXIDR</td>
</tr>
</tbody>
</table>

Table 4: CXIDR_RESPONSE_RULES

<table>
<thead>
<tr>
<th>RESPONSE_ID</th>
<th>DESCRIPTION</th>
<th>TYPE</th>
<th>SOURCE_MOD</th>
<th>DEPLOYMENT_MOD</th>
<th>PRIORITY</th>
<th>RELATED LAYER</th>
<th>COST</th>
<th>RESPONSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>02X</td>
<td>YYY</td>
<td>SHORT</td>
<td>SRC</td>
<td>SRC</td>
<td>1</td>
<td>APPLICATION</td>
<td>2</td>
<td>CXIDR</td>
</tr>
<tr>
<td>02X</td>
<td>SRC</td>
<td>SRC</td>
<td>SRC</td>
<td>SRC</td>
<td>1</td>
<td>APPLICATION</td>
<td>4</td>
<td>CXIDR</td>
</tr>
<tr>
<td>02X</td>
<td>SRC</td>
<td>SRC</td>
<td>SRC</td>
<td>SRC</td>
<td>2</td>
<td>NETWORK</td>
<td>5</td>
<td>CXIDR</td>
</tr>
<tr>
<td>02X</td>
<td>SRC</td>
<td>SRC</td>
<td>SRC</td>
<td>SRC</td>
<td>5</td>
<td>PHYSICAL</td>
<td>6</td>
<td>CXIDR</td>
</tr>
</tbody>
</table>

Information of CXIDR_TEMPORARY_DATABASE table and CXIDR_RESPONSE_RULES table are shown in details in Table 3 and Table 4. In the following figure...
(Fig. 8), implementation of response selection algorithm which is coded in Java language using NetBeans software is shown. Here, ojdbc6 JAR library is used for this implementation process. The SQL query of response selection algorithm is shown below Fig. 8.

```java
public class CXIDR {
    // Java code application logic here
    public static void main(String[] args) {
        Connection con = null;
        Statement stmt = null;
        ResultSet rs = null;
        // Resultset rs = null;
        try {
            Class.forName("oracle.jdbc.driver.OracleDriver");
            con = DriverManager.getConnection("jdbc:mysql://localhost:3306/");
            stmt = (Statement) con.createStatement();
            // rs = stmt.executeQuery("SELECT * FROM CXIDR_TEMPORARY_DATABASE"
            while (rs.next()) {
                // "SELECT UNIQUE " + "FROM CXIDR_TEMPORARY_DATABASE, CXIDR_RESPONSE_RULES"
                // WHERE CXIDR_TEMPORARY_DATABASE.PRIORITY =
                // CXIDR_RESPONSE_RULES.PRIORITY AND
                // CXIDR_TEMPORARY_DATABASE.RELATED_LAYERS =
                // CXIDR_RESPONSE_RULES.RELATED_LAYERS;
            }
        }
    }
}
```

**Fig. 8** Implementation of Algorithm in Netbeans

**SQL Query:**

```
SELECT UNIQUE
CXIDR_RESPONSE_RULES.PRIORITY,
CXIDR_RESPONSE_RULES.RESPONSE
FROM CXIDR_TEMPORARY_DATABASE,
CXIDR_RESPONSE_RULES
WHERE
CXIDR_TEMPORARY_DATABASE.PRIORITY=
CXIDR_RESPONSE_RULES.PRIORITY AND
CXIDR_TEMPORARY_DATABASE.RELATED_LAYERS=
CXIDR_RESPONSE_RULES.RELATED_LAYERS;
```

After running the upper SQL query in Oracle the following query result (Fig. 13) is displayed. Using this response from the SQL query some conditions will be applied upon them with respect to cost according to the response selection algorithm and response with the minimum cost will be deployed by the Response Deployment Module.

Table 5. Result of SQL Query

<table>
<thead>
<tr>
<th>PRIORITY</th>
<th>RESPONSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>ABORT</td>
</tr>
<tr>
<td>2</td>
<td>BLA BLA</td>
</tr>
<tr>
<td>3</td>
<td>DO SOMETHING</td>
</tr>
</tbody>
</table>

In Fig. 9 application of Snort is shown. We have implemented Snort with the help of Kiwi Syslog Server which is used as a Graphical User Interface (GUI) for showing the analysis of incoming and outgoing packets. From this sort of analysis intrusions can be easily detected.

**Fig. 9** Packet Analysis Using Snort.

Implementation of Received Signal Strength (RSS) is displayed in Fig. 10. It is implemented using NS-3. Here, Signal to Noise Ratio or SNR is counted. We have used this value as RSS in our system. The SNR value is calculated and then converted into SNR db (Decibel). In Fig. 10 a table showing the packet rate and RSS value in decibel is displayed.

**Fig. 10** Implementation of RSS

<table>
<thead>
<tr>
<th>Packet</th>
<th>RSS(Decibel)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>119.35</td>
</tr>
<tr>
<td>5</td>
<td>123.077</td>
</tr>
<tr>
<td>7</td>
<td>257.859</td>
</tr>
<tr>
<td>10</td>
<td>257.859</td>
</tr>
<tr>
<td>100</td>
<td>257.859</td>
</tr>
</tbody>
</table>

Table 6. Packet Rate and Corresponding RSS value

5. Conclusions

In this paper, we have introduced a novel IDS framework based on honey pot approach that combined multiple intrusion detection sources and utilized it in cellular environment. We have also introduced a cost-effective response selection module across various layers. We showed that our proposed CXIDR methodology is capable to outperform the conventional one. In future, we shall
enhance CXIDR further and add some other security related features with the current system. There may be some other works of implementing the algorithm and building a response deployment module so that present CXIDR can face the upcoming challenges in the future. The methodologies we proposed in this article are based on new technologies and it might be a complete scheme to safeguard our precious data and devices in cellular environment.

References