# An Implementation of IDATA - Intercept Detection Algorithm for Packet Transmission in Trust Architecture

GJCST Classification (FOR)
C.2.1. E.3

Dr. S. N. Panda[1], Gaurav Kumar[2]

***Abstract*** **- World is growing with the emerging technologies. The computer networks and packet transmission systems are also growing in parallel, hence to manage and provide security to packet, a secured system is required. Networks seize or simply intercept is one of the challenges in the fast growing world of Cyber Crime. The network establishments are facing various types of threats on routine basis. To efficiently transmit information across a network, there is need of an improved and reliable architecture. An intrusion or intercept refers to an active sequence of events that deliberately try to cause harm, such as rendering system unusable, accessing unauthorized information, or manipulating such information. Security professionals may want to have Intercept Detection Systems record information about both successful and unsuccessful attempts so that security professionals will have a more comprehensive understanding of the events on their networks. The intercept detection systems should be developed with utmost care to avoid any natural or intentional attempts. Moreover, the packet encryption algorithm should be developed in such a way so that cracker is not able to change even a single bit in the confidential data. This paper illustrates the implementation of IDATA - An Intercept Detection Algorithm in the Trust Architecture including the development and execution of a secured and efficient encryption algorithm for efficient and secured data packet transmission.**

*Keywords* - Cyber Security, Trust Architecture, Intercept Detection, Intrusion Detection, Trust Architecture, E-Transactions, Interception Analysis and Forensics, Packet Encryption, Packet Decryption, Packet Transmission

## I. INTRODUCTION

Now days, the commercial as well as Defense Applications are facing frequent threats from different source and obviously such highly sensitive applications of public and national interest needs highly secured and consistent architecture so that packets can be transmitted in the network without any peril. Trust is considered as the footing of the relationship which is established by a business organization with their customers, vendors, and employees. All Trust Architectures and Intercept detection technology are not effective. These neither provided security to packet formation nor giving any security during transmission. All Trust Architecture developed till now doesn't provide absolute security and significant features. The VAN sometimes paralyzed and giving a great scope to the intruders/interceptors and other cyber criminals either to damage or alter or misuse the packets during transmission. Most of the fund transfer systems, EDI systems, business applications are using emerging technologies and exposed to vulnerability increases tremendously. Moreover, the cryptographic algorithms used during packet formation and transmission are sometimes responsible for vulnerabilities. Trust is the establishment of confidence that something will or will not occur in a predictable or promised manner. The enabling of confidence is supported by identification, authentication, accountability, authorization, and availability. To develop the trust between multiple parties, a set of principles or rules is to be offered so that the security of the entire model can be improved. According to the ITU-T X.509, Section 3.3.54, trust is defined as: "Generally an entity can be said to 'trust' a second entity when the first entity makes the assumption that the second entity will behave exactly as the first entity expects."

## II. INTERCEPT DETECTION SYSTEMS (IDS)

An intrusion-detection system (IDS) refers to the tools, methods, and resources to help identify, assess, and report unauthorized or unapproved network activity. The intrusion detection part of the name is a bit of a misnomer, as an IDS does not actually detect intrusions - it detects activity in traffic that may or may not be an intrusion. Intrusion detection is typically one part of an overall protection system that is installed around a system or device - it is not a stand-alone protection measure.

*About[1]- Professor & Principal, Regional Institute of Management and Technology [RIMT],Mandi Gobindgarh, Punjab E-mail: panda.india@gmail.com*
*About[2]- Sr. Lecturer, Computer Applications Chitkara Institute of Engineering and Technology, Rajpura, PunjabE-mail: kumargaurav.in@gmail.com*

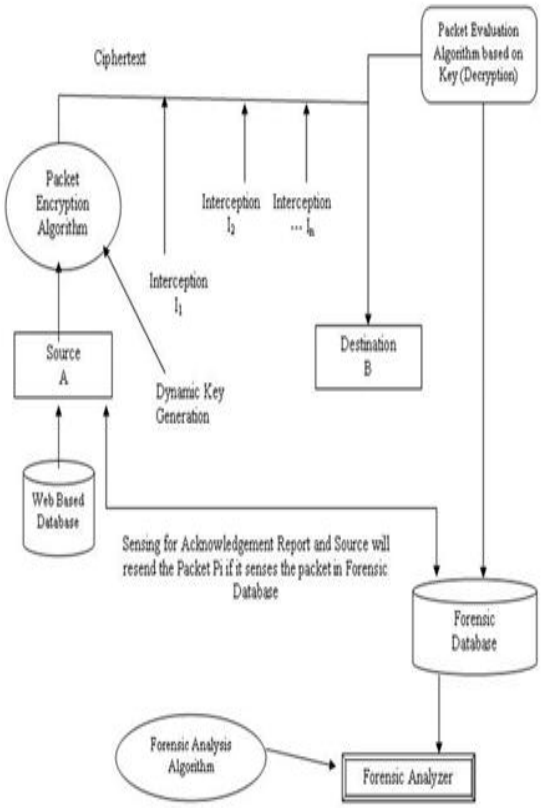



Figure 1 : The Proposed Trust Architecture for Intercept Management

1) *Algorithmic Approach*

**Step 1:** Initialize & Activate Packet $P_i$ at Source $S_i$ for transmission to Destination $D_i$

**Step 2:** Packet Encryption Module $PE_k$ based on Dynamic Key k Generation, once the Packet moves from Source $S_i$

$C_i := PE_k (P_i)$

**Step 3:** Transmission of Encrypted Packet $C_i$ using specified Path/Route $R_i$

$C_i \rightarrow D_i [R_i]$

**Step 4:** Packet Authentication on Decryption

The most common types of threats fall into categories such as:

- Actual or attempted unauthorized probing of any system or data
- Actual or attempted unauthorized access
- Introduction of viruses or malicious code
- Unauthorized modification, deletion, or disclosure of data
- Denial of service attacks

## III. PROPOSED ALGORITHMS

The proposed architecture consists of various phases which will include algorithms for encryption and decryption of data packet alongwith the technique to analyze the overall interception patterns.

IF ( $C_i$ = $PD_k (C_i)$ // Packet Decryption Module $PD_k$ to decrypt the packet at destination
BEGIN
DEST [i] := $PD_k (C_i)$
Successful Delivery of Packet
ACK sent to Source $S_i$ // Acknowledgement ACK is delivered to Source in case of Success
END
ELSE
BEGIN
A record will be inserted in the Forensic Database. The Interception Table will consist of the Structure (Id, Interception Type, Timestamp of Interception). // Acknowledgement ACK is sent to Forensic Database in case of Failure Attempt
Source $S_i$ senses the Forensic Database.
Select All Records from Forensic Database
IF (true)Then
print "Failure Delivery, Retransmit the packet"
GOTO Step 1
Update Forensic Analyzer Database for taking remedial actions.
END

**Step 5: Forensic Analyzer**

Retrieve Records for analysis of interceptions. In the proposed architecture, an extended flavor of link level encryption will be used to encrypt the entire data packet.The packet encryption algorithm at the originating site encrypts the entire packet including the packet header and

1) *Encryption Algorithm*

**Step 1:** Activate and Initialize the Packet $P_i$
**Step 2:** Generate a Random Key $K_R$ by analyzing number of 1s in Packet.
(a) Develop a routine to count bits in the Data Packet
(b) Set N := Count($P_i$) // Count Number of 1's in the Data Packet.

2) *Encryption Algorithm*

**Step 1:** Activate and Initialize the Packet $P_i$
**Step 2:** Generate a Random Key $K_R$ by analyzing number of 1s in Packet.
(a) Develop a routine to count bits in the Data Packet
(b) Set N := Count($P_i$) // Count Number of 1's in the Data Packet.

## IV. DECRYPTION AND INTERCEPT DETECTION ALGORITHM

A decryption algorithm at the destination site will check the entire encrypted packet. The received packet will be of specific format and structure in which key is given. By

*Analyze the type $T_i$ of Intercept*

*Perform remedial stroke for avoiding the stored interception type*

The proposed architecture consists of various phases which will include algorithms for encryption and decryption of data packet alongwith the technique to analyze the overall interception patterns.

## V. Packet Encryption Algorithm

At the initial stage, the data packet will be transmitted from source to destination over transmission media using efficient cryptographic algorithm to encrypt the entire packet. Cryptography is the process used to make a meaningful message appear meaningless. An algorithm is a set of rules or procedures used to scramble, or encrypt the plaintext to produce Ciphertext. The algorithm applies a key to text.Encryption is the procedure that guarantees secrecy of the data exchanged. Any encryption algorithm depends on some key, and keys are normally generated during authentication phase, so the two phases are strictly connected. provides it a new header. This readable new header also includes a dynamic key-id. The key-id controls the behavior of encryption and decryption mechanism. It specifies the information as the encryption algorithm, the encryption block size, the error checking code and lifetime of the key.

(c) Set $K_R$ :=N // Store N in Random Number $K_R$

**Step 3: Apply XOR (Exclusive-OR) Operation**

(a) Set $E_K := P_i \oplus_R$

(b) The Encrypted Packet $E_K$ is generated using XOR Operation.

(c) Set $PE_K := E_K$ // Utilize $E_K$ as Encrypted Packet

**Step 4: Packet equipped for Transmission**

analyzing the structure of encrypted packet, the location of key will be accessed and the packet can be decrypted. In case of interceptions at the transmission line, the details of such attempts will be stored in the web based databases so that interception points and sources can be identified. In case, there is an interception and packet is not matched after decrypting the Ciphertext $C_p$, a record will be inserted in the forensic database. The pattern/behavior of intercepts will be analyzed using a forensic analyzer. In case of successful decryption and transmission of packet, an acknowledgement will be transmitted to the web based

1) *Algorithm*

**Step 1:** Receive the Encrypted Packet $PE_K$

**Step 2:** Check the Front $PF_i$ and Rear End $PR_i$ of Packet

if ($PF_i = PR_i$)

Accept PFi

Set $K_R := PF_i$

else

goto Step 5

**Step 3:** Generate the Binary Equivalent of $K_R$

$PB_i$ = Binary($K_R$)

**Step 4:** Perform XOR Operation

if ($PB_i = PE_K$)

Decryption Successful Accept the Packet Else goto step 5

Step 5: Insert the Record of Corrupt Packet in Forensic Database

## VI. Implementation and Simulation

The cryptographic algorithm based on XOR logic gate is implemented and simulated and tested using the following

**MS Windows Platform**

Windows XP,

Turbo C IDE

**Linux Platform**

Fedora 11

gcc

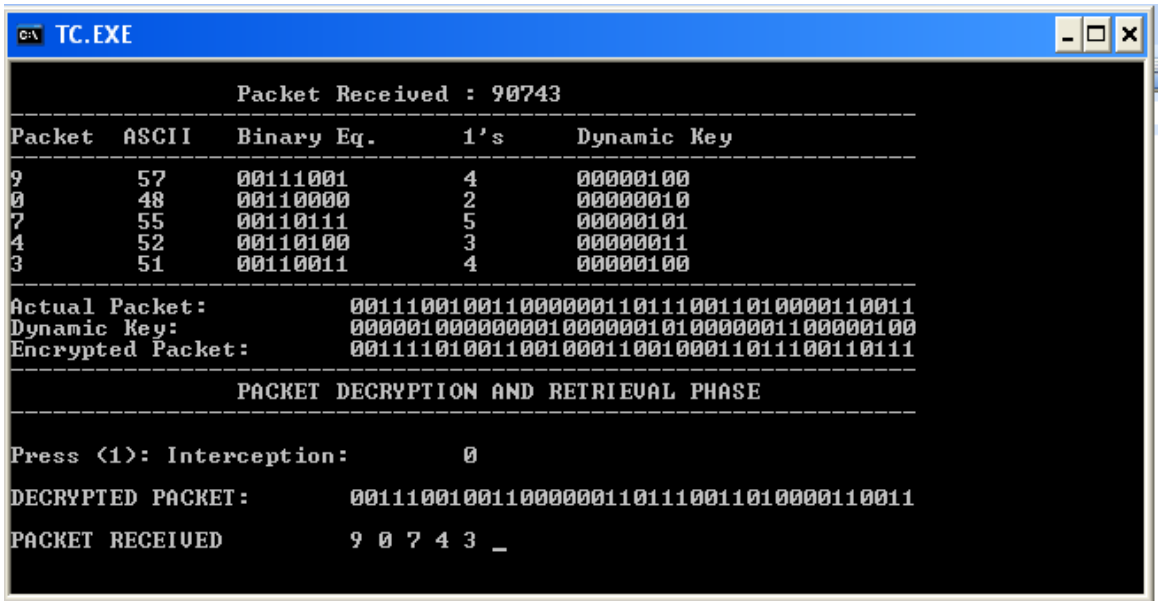
**SOURCE CODE IN C**

```
int a[40], b[40], c[40], decrypted[40], choice;
char d[40];
FILE *key, *source, *f1, *f2;
char buf[5];
void main()
{
FILE *fp;
int random, i;
clrscr();
fp=fopen("packet.txt","r");
source=fopen("sourcebin.txt","w+");
key=fopen("keybin.txt","w+");
randomize();
random=rand()%5;
fseek(fp,5*random,0);
fread(buf,5,1,fp);
printf("\n\t\tPacket Received : %5s\n",buf);
printf("Packet\tASCII\tBinary Eq.\t1's\tDynamic Key\n");
buf[5]='\0';
database where the source site can verify the delivery of
message.
for (i=0;i<5;i++)
{
delay(50);
printf("\n%c\t%3d\t",buf[i], buf[i]);
decbin(buf[i]);
printf("\t");
saveresults_a(buf[i]);
printf("%d",countdecbin(buf[i]));
printf("\t");
decbin(countdecbin(buf[i]));
saveresults_b(countdecbin(buf[i]));
}
fclose(source);
fclose(key);
xor_op();
delay(200);
printf("\t\t");
textattr(128+10);
decrypt_packet();
for (i=0;i<40;i++)
{
if (a[i]!=decrypted[i])
{
```

```
printf("\n\t\t\t");
textattr(128+YELLOW);
cprintf("Interception ! Packet sent to Forensic Database");
}
}
getch();
}
int decbin(int number)
{
int x, y;
x = y = 0;
for(y=7; y>=0; y--)
{
  x = number / (1 << y);
  number = number - x * (1 << y);
  printf("%d",x);
}
return 0;
}
int saveresults_a(int number)
{
int x, y;
x = y = 0;
for(y=7; y>=0; y--)
{
  x = number / (1 << y);
  number = number - x * (1 << y);
  fprintf(source, "%d ", x);
}
return 0;
}
int saveresults_b(int number)
{
int x, y;
x = y = 0;
for(y=7; y>=0; y--)
{
  x = number / (1 << y);
  number = number - x * (1 << y);
  fprintf(key, "%d ", x);
}
return 0;
}
int countdecbin(int number)
{
int x, y, count=0;
x = y = 0;
for(y = 7; y >= 0; y--)
{
  x = number / (1 << y);
  number = number - x * (1 << y);
  if (x==1)
  {
  count++;
  }
}
return count;
}
int xor_op()
{
int i=0, j=0;
f1=fopen("sourcebin.txt","r");
f2=fopen("keybin.txt","r");
do
{
   fscanf(f1, "%d", &a[i]); i++;
} while (i<40);
do
{
   fscanf(f2, "%d", &b[j]); j++;
} while (j<40);
printf("Actual Packet: \t\t");
for (i=0;i<40;i++)
{
delay(50);
printf("%d",a[i]);
}
printf("\nDynamic Key: \t\t");
for (i=0;i<40;i++)
{
delay(50);
printf("%d",b[i]);
}
fclose(f1);
fclose(f2);
for (i=0;i<40;i++)
{
if (a[i]==b[i])
c[i]=0;
else
c[i]=1;
}
printf("\nEncrypted Packet:\t");
for (i=0;i<40;i++)
printf("%d",c[i]);
return 0;
}
int decrypt_packet()
{
int i=0;
int p;
char bb[40];
char xx='1', yy='0';
int k=0, j=8, x;
for(i=0; i<40; i++)
{
if (c[i]==b[i])
{
d[i]=yy;
decrypted[i]=0;
}
else
{
d[i]=xx;
```

```
decrypted[i]=1;
}
}
d[40]='\0';
printf("\n");
textcolor(WHITE);
textbackground(BLUE);
cprintf("Press (1): Interception:");
printf("\t");
scanf("%d",&choice);
if (choice==1)
{
decrypted[10]=2;
d[10]='.';
}
printf("\nDECRYPTED PACKET:\t");
for (i=0;i<40;i++)
{
delay(50);
printf("%c",d[i]);
}
printf("\n\nPACKET RECEIVED\t\t");
i=0;
while(i<=40)
{
for (p=0,i=k;i<j;i++,p++)
{
bb[p]=d[i];
}
textattr(128+10);
cprintf("%c ",bin2dec(bb));
k=k+8;j=j+8;i=i+8;
}
return 0;
}
int bin2dec(char *bin)
{
int b, k, m, n;
int len, sum = 0;
len = strlen(bin) - 1;
for(k = 0; k <= len; k++)
{
n = (bin[k] - '0'); // char to numeric value
for(b = 1, m = len; m > k; m--)
{
// 1 2 4 8 16 32 64 ... place-values, reversed here
b *= 2;
}
// sum it up
sum = sum + n * b;
}
return(sum);
}
```

## VII. RESULTS OBTAINED




## VIII. CONCLUSION

The business, defense and government applications are required to be deployed in a highly secured and confidential environment which needs secured architecture as well as an efficient method of data packet encryption. Different types of networks are in front of number of threats from increasing intercepts originating from various sources. To secure these applications from unauthorized and illegal access, there is need to secure the network from multiple interceptions using efficient algorithms. The implementation demonstrated in this paper illustrates an efficient algorithm based on Exclusive-OR operation which is a unique method

to encrypt any data packet traveling in the network. Using this method, encryption, decryption and traveling of packet can be performed effectively without any complexity. Moreover, the forensic database module keeps track of every invalid or unacceptable decrypted packet. With the prior information of unauthorized access of records in the database, the behavior of intercepts can be analyzed to avoid such attempts in future.

## IX. References


1) Cochavy, Baruch, Method of efficiently sending packets onto a network by eliminating an interrupt, US Patent 5797039 Issued on August 18, 1998
2) Dimitris M. Kyriazanos, Neeli R. Prasad, Charalampos Z. Patrikakis, A Security, Privacy and Trust Architecture for Wireless Sensor Networks, 50th International Symposium ELMAR-2008, 10-12 September 2008, Zadar, Croatia
3) Dr. S. N. Panda, Gaurav Kumar, "IDATA – An Effective Intercept Detection Algorithm for Packet Transmission in Trust Architecture" (POT-2010-0006), selected for publication in IEEE Potentials ISSN: 0278-6648.
4) Dr. S. N. Panda, Gaurav Kumar, "Effective Implementation Of Intruder Detection Trust Architecture Using XOR Logic Gate Cryptographic Technique" published in Journal of Ultra Scientist of Physical Sciences, Bhopal, ISSN 0970-9150, Vol. 22 No. 1, April 2010.
5) Donna Andert, Robin Wakefield, and Joel Weise, Professional Services Security Practice, Sun BluePrints™ OnLine - December 2002, Trust Modeling for Security Architecture Development
6) Security, Encryption, Acceleration, http://www.networkintercept.com
7) Youlu Zheng, Shakil Akhtar, Networks for Computer Scientists and Engineers, Oxford University Press, 2009
8) Carl Endorf, Eugene Schultz and Jim Mellander, Intrusion Detection & Prevention, McGraw-Hill, 2004