Covert Channel Detection and Analysis System Based on Data Mining

YAO WANG, MINGZENG HU, BIN LI

Research Center of Computer Network and Information Security Technology,
Harbin Institute of Technology, 92 Dazhi Street, Nangang District,
Harbin 150001, Heilongjiang, CHINA

Abstract: - Covert channels and tunneling approaches are becoming a severe threat to information security. Penetration tools are employed to transit sensitive information through authorized streams. Since many current solutions are based on expert's experiences or latter-wit, a self-learning detection and analysis system is starved for. A data mining framework for Covert Channel Detection and Analysis System (CCDAS) is presented in this paper. It utilizes protocol analysis method to reassemble each network connections according to all kinds of protocol specifications, and apply data mining programs to construct features that can accurately distinguish the abnormal behaviors of covert channels from the normal activities. The main components of CCDAS (namely Protocol Analyzer, Feature Constructor and Class Identifier) work together and detect various emerging covert channels automatically.

Key-Words: - information security, covert channel, data mining

1 Introduction

Computer networks play an increasingly vital role in information society. To reinforce their security, numerous protection equipments such as Firewall or IDS are employed. However, many evasion methods are devised to access external services within the internal network or access internal resources from the external network arbitrarily[1][2]. These methods allow the opening of communication channels (e.g., covert channels) to use streams authorized by the security policy to transit arbitrary data whose traffic is not allowed or thought of. All these make the network information security issues outstanding day by day.

1.1 Definitions of covert channel

A covert channel is defined as any communication channel that can be exploited by a process to transfer information in a manner that violates the system's security policy [3]. By convention, covert channels are classified as storage channels in which the sender affects the value of some storage entity that can be observed by the recipient or timing channels in which the sender affects the recipient's perception of the time required to carry out some action.

In this paper, we focus on covert channels in real network environment, and information penetrations take place by making use of the network packets as their covertures. We name the technologies which transfer sensitive information through covert channel as Covert Channel Penetration Technology (CCPT).

1.2 Related work

Covert channels are discussed in a wide number of papers. Girling [4] first analyzes covert channels in a network environment. His work demonstrates the real examples of the bandwidth possibilities for simple covert channels in LANs. Wolf [5] presents the fact that encryption as the basic mechanism of LAN security can't ensure the proper blocking of unauthorized information via covert channels.

A general analysis paradigm to covert channels is presented in [6] with two forms of analysis: Shared Resource Matrix and Information Flow Analysis on analyzing a system for covert channels. Serious security flaws in the TCP/IP protocol suite with details on a variety of attacks are described in [7]. Besides identifying threats, it also presents broad-spectrum defenses such as encryption. In order to protect network against security vulnerabilities, redundancy is introduced into the protocol specifications, however, as discussed in [8], redundancy in communication elements is a main carrier for covert transmission. The OSI network model is employed as a basis for characterizing system elements having potential to be used for

covert channels. Recently, many new practical approaches are presented. Muhammad et al. [9] present a dynamic simulation model and testing techniques for security protocol verification, which aims at minimizing the flaws in simulation and increase the efficiency of protocol verification procedure. Chanana et al. [10] present a new context-based information retrieval (IR) system. By representing the context based on the type of information, it overcomes major limitations of the keyword-based IR systems. As described in [11], authorized but malicious transactions can make a database useless by impairing its integrity and availability. The proposed solution offers the ability to detect misuse and subversion through the direct monitoring of database operations inside the database host, providing an important complement to host-based and network-based surveillance. But how to detect the violation in a large-scale network environment is not taken into account.

So how to discover covert channels exactly and in time in a real-time network environment has become a great challenge to network security researchers. The objective of this paper is to present a data mining framework to detect and analyze various covert channels automatically.

The rest of the paper is organized as follows. In Section 2, we describe Characteristics of CCPT firstly and classify current CCPTs from the methodology point of view subsequently. According to this classification, we introduce three kinds of detection methods to analyze various covert channels in Section 3. We then present a framework of CCDAS in Section 4 and evaluate the experiment results obtained from this prototype system in Section 5. In Section 6, we draw a conclusion and discuss our future work.

2 Classification of CCPT

2.1 Characteristics of CCPT

The three intrinsic characteristics of CCPT are: Penetration Destination (PD), Penetration Method (PM) and Penetration Content (PC). For each kind of CCPTs, there exists a characterization triple <PD, PM, PC>. The descriptive model of CCPT is to transfer Penetration Content to Penetration Destination by Penetration Method, namely PM (PC)→PD.

2.2 Classification

We classify CCPTs with their intrinsic characteristics by analyzing their implementation mechanisms. The purpose of CCPTs is to transit arbitrary data stealthily, so they try to conceal their three sensitive characteristics from being detected. The fundamental methods that CCPTs adopted are encryption and/or transformation, which trying to wrap the penetration destination and content.

According to different modification targets, CCPTs are divided into three kinds of types: Destination Modification, Method Modification and Content Modification. They are described as follows:

(1) Destination Modification (proxy chains/intermediary distributed servers)

It usually changes the fixed position of address in packets with proxy technology. Simple modifications transfer the address from IP header to TCP data section such as HTTP proxy or SOCKS proxy; and complicated ones include setting up proxy chains or intermediary distributed servers.

(2) Method Modification (converted/special protocols)

It adopts converted or special protocols instead of routine protocols (e.g. HTTP or SSL) to hide the traces in the mutual course.

(3) Content Modification (substitution/ encryption)

Data streams are coded or encrypted so that matching the content directly becomes unfeasible.

Therefore, effective detection techniques must be studied to defeat these CCPTs.

3 Detect and Analyze Covert Channel

There are three kinds of detection methods used cooperatively to detect various covert channels.

(1) The feature recognition method involves constructing and updating a feature database and matching the known features in the network data streams.

(2) The protocol analysis method focuses on reassembling well-known protocols according to the RFC's specifications and presenting protocol anomalies or violations to decision engine.

(3) The behavior detection method involves using statistical methods to detect abnormal network data streams.

3.1 Feature recognition

The data stream transiting through covert channels contains some fingerprints such as a specific URL or an odd port number. All these can be used as the identity of CCPTs. So how to recognize

these fingerprints from the network data streams monitored is the linchpin of covert channel detection.

With the RIPPER, a rule learning algorithm [12], a well-defined rule set shown in Table 1 can be induced to construct the corner stone of feature database.

RIPPER rule	Meaning
CCPT1:-Con_num(s→d) ≥5,dur_time < 5s, out_data_size > 1000KB	If the number of connections from s to d is at least 5 during less than 5 seconds, and the size of outbound data is greater than 1000k bytes, then this connection is CCPT1.
CCPT2:-dst_port=1111, in_data_size = 100B	If the destination port is 1111, and the size of inbound data is always 100 bytes, then this connection is CCPT2.
……	……
Normal:-true.	If none of the above, then this connection is "normal".

Table 1 RIPPER Rules for Classifying CCPTs

```
Training Period:

Begin
  if (s->d)
  {
    Learning_Connections[s->d] += 1;
    Record ( n, t, a );
    if (( n > 5 ) and ( t < 3 ) and ( a > 1000 ))
    {
      Learning_Counter[s->d] += 1;
      Reset ( n, t, a );
    }
  }
  Threshold := Learning_Counter[s->d]/Learning_Connections[s->d];
End;

Executing Period:

Begin
  if (s->d)
  {
    Running_Connections[s->d] += 1;
    Record ( n, t, a );
    if (( n > 5 ) and ( t < 3 ) and ( a > 1000 ))
    {
      Running_Counter[s->d] += 1;
      Reset ( n, t, a );
      If ((Running_Counter[s->d]/Running_Connections[s->d])>Threshold)
        Raise_Alarm("Connection of 's->d'");
    }
  }
End;
```

Fig.1 Pseudocode for Statistical Detection

3.2 Protocol analysis

This approach involves searching for protocol anomalies or violations in the network environment. It must be carefully designed concerning all states happening in communication and appropriate operations in each step according to protocol specifications or protocol's normal usages.

For example, when HTTP server responses per HTTP request, a Hostname header field must exist when the HTTP/1.X protocol is used, however, some CCPTs modify this field to transfer other arbitrary data, so the well-designed protocol analysis can detect this violation following most of the RFC's specifications especially practical usages.

3.3 Behavior detection

To build a covert channel, a great deal of prophase work is needed. For example, a telnet-like covert channel will generate a very small but frequent amount of data at the beginning, and requests or responses are sent too frequently or are sent on a curious static interval. These unique behaviors can be employed to detect unknown CCPTs.

Statistical approaches are involved to identify the abnormal behaviors. A threshold based on the reference profile is calculated and if an attribute value exceeds the threshold, the corresponding event will be recorded to judge.

This method involves checking quantitatively the attribute values of specific data streams during training period to generate a reference profile. When the CCDAS runs in an executing period, it dynamically checks the reference profile against attribute values of real-time data streams using statistical operations and if a difference upper than the pre-setting X% threshold, an alarm is triggered.

Let's take a look at the illustration in Fig.1. When a connection occurs for more than five times during three seconds between the same source address and the same destination address and the data transferred beyond one megabyte, maybe it is a potential covert channel on its establishing state. Define connection from source address s to the destination address d as s→d, and time span since the beginning of s→d is t, record the amount of data a from s to d and the number of connections n occurring on each s→d communications.

4 Framework of CCDAS

This framework adopts a hierarchical structure formed by independent modules: Packet Capturer (PC), Protocol Analyzer (PA), Feature Constructor (FC), Class Identifier (CI) and Decision Engine (DE). They will be introduced separately combining Fig.2.

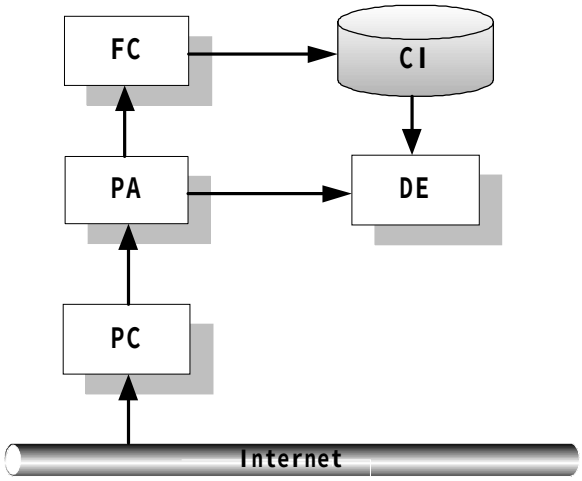

Fig.2 Data Process of CCDAS

4.1 Packet Capturer (PC)

In order to construct an accurate feature set, a sufficient amount of meaningful network data are needed as the input of data process. PC utilizes the kernel capture mechanism of the operating system that CCDAS affiliated with to gather the raw data in data link layer by setting the network interface to promiscuous mode. In the system, we utilize Libpcap to interact with kernel driver directly to accomplish the data collection work.

4.2 Protocol Analyzer (PA)

PA acts as the network protocol stack. It deals with the raw data stream gathered by PC according to different protocol specifications. PA refers the formatted data records to FC for analysis. PA's perfectibility directly determines both the system's ability for dealing with different protocols and the system's security. It is also closely related to FC, whether FC succeeds or not depends on the recover degree of protocols.

During the analysis process, several key attributes such as source IP, source port, destination IP, destination port, data content, data size and protocol type are involved to form the formatted data records.

4.3 Feature Constructor (FC)

The features are constructed based on frequent patterns computed from the formatted data records provided by PA which capture the actual behaviors of covert channels in the forms of statistical summaries. With behavior detection method described in section 3.3, data mining programs are then applied to the records to compute the frequent patterns, which in turn describing the abnormal behaviors so that additional features for the formatted data records are constructed.

The key to FC is based on the mining of frequent patterns, so timers and counters should be allocated to record the connection time or connection counts for calculating the frequent patterns over the threshold in order to find the valuable association rules and frequent episodes.

4.4 Class Identifier (CI)

CI is a crucial component of CCDAS. It adopts the feature extraction method introduced in section 3.1 to build a class database. Depending on the RIPPER rules, it can identify all known CCPTs and submit them to DE as the basis of assertion. Besides this, CI receives the potential features from FC to induce rule sets for building corresponding classifications. By enriching and revising the CI component constantly, more and more unknown CCPTs will be detected.

4.5 Decision Engine (DE)

DE disposes the events received from CI and PA to seek potential covert channels and takes the corresponding measures as soon as possible. The procedure includes two aspects:

(1) Classify known or forecasted covert channel events from CI into different ranks (e.g. suspicious or severe) and take corresponding actions.

(2) Deal with protocol anomalies or violations from PA and take corresponding responses.

5 Experiment and Evaluation

In this section, we implement the prototype of CCDAS in the lab environment and evaluate the experiment results obtained from a training period and a running period, each lasting five days respectively.

5.1 Experiment results

We take five kinds of CCPT tools as our experiment objects. To avoid the unnecessary offence, we omit the true name of these tools. The experiment is executed in two steps: Training Period and Running Period. In each period, the five tools are tested for 5 days and both the total test times and the detection times of each CCPT tool are recorded in every single day. The experiment results are illustrated in Table 2.

Step	Training Period									
Day	1		2		3		4		5	
	T	D	T	D	T	D	T	D	T	D
ccpt1	41	25	47	27	52	37	49	30	49	33
ccpt2	38	15	47	27	48	20	48	17	46	17
ccpt3	42	12	49	12	48	13	48	15	45	15
ccpt4	42	10	49	23	49	25	48	22	46	36
ccpt5	40	32	47	33	48	46	47	47	45	44
Step	Running Period									
Day	1		2		3		4		5	
	T	D	T	D	T	D	T	D	T	D
ccpt1	44	32	43	30	45	40	34	28	42	27
ccpt2	43	23	42	23	45	25	35	7	42	17
ccpt3	43	22	43	18	48	24	34	29	42	39
ccpt4	43	22	43	21	47	35	35	27	42	36
ccpt5	41	41	41	41	45	45	34	32	42	41
T: Total test times					D: Detection times					

Table 2 Test Results of Five CCPT Tools

5.2 Evaluation

The average detection rate in the Training Period and the Running Period are calculated respectively and shown in Fig.3. By comparison, we can see the average detection rate increased about 10% in the Running Period and the detection rate of CCPT3 is promoted even more than 35%.

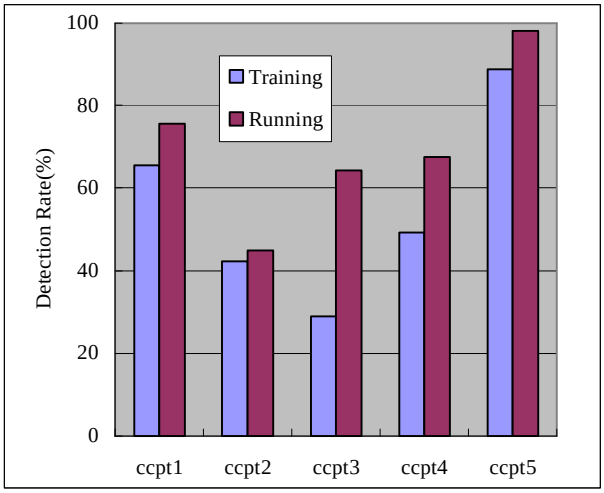

Fig.3 Comparison of Detection Rate Between Training and Running Period

6 Conclusions and Future Work

In this paper, the framework of CCDAS is presented in the real network environment. The intrinsic characteristics of CCPT are abstractly extracted and according to which the CCPTs are classified accurately and effectively. Feature extraction, protocol analysis and behavior detection are integrated into CCDAS so that it can detect and analyze both known and potential CCPTs with improving detection rate automatically.

Future work will include deploying distributed network architecture with detection sensors and optimizing the performance of abnormal behavior detection in the real-time network environment.

References:

[1] C. H. Rowland, Covert channels in the TCP/IP protocol suite, *Tech. Rep. 5, Peer Reviewed Journal on the Internet*, 1997

[2] John Giffin, Covert Messaging through TCP Timestamps, *Proceedings of PET2002*, pp. 194-208, San Francisco, 2002

[3] National Computer Security Center, Department of Defence Trusted Computer System Evaluation Criteria, DoD 5200.28-STD, 1985

[4] C. G. Girling, Covert channels in LAN's, *IEEE Transactions on Software Engineering*, Vol.SE-13, No.2, 1987

[5] M.Wolf, Covert channels in LAN protocols, *Proceedings of the Workshop on Local Area Network Security*, 1989, pp. 91-102.

[6] John Mchugh, Covert Channel Analysis, *Formal Methods in Computer Security*, 1999

[7] S. M. Bellovin, Security problems in the TCP/IP protocol suite, *Computer Communication Review*, Vol.19, 1989, pp. 32-48.

[8] T. Handel and M.Sandford, Hiding data in the OSI network model, *Proceeding of First International Workshop on Information Hiding*, Cambridge, 1996

[9] S. Muhammad, R. Guha and Z. Furqan, A Dynamic Simulation Model and Testing Techniques for Security Protocol Verification, *WSEAS Transactions on Computer*, Issue 5, Vol.3, 2004, pp. 1226-1231.

[10] V. Chanana, A. Ginige and S. Murugesan, A New Context-Based Information Retrieval System, *WSEAS Transactions on Computer*, Issue 4, Vol.3, 2004, pp. 873-879.

[11] U. T. Mattsson, A Practical Implementation of a Real-time Intrusion Prevention System for Commercial Enterprise Databases, *WSEAS Transactions on Communications*, Issue 1, Vol.3, 2004, pp. 179-184.

[12] W. W. Cohen, Fast effective rule induction, *Proceeding of the 12th International Machine Learning Conference*, Lake Taho, CA, 1995