IOSR Journal of Computer Engineering (IOSRJCE)
ISSN: 2278-0661, ISBN: 2278-8727 Volume 7, Issue 2 (Nov-Dec. 2012), PP 35-40
www.iosrjournals.org

Securing Data using Pre-filtering and Traceback Method

G.Vaithiyanathan[1], V.Sridevi[2], S.Supriya[3], B.Divya[4], Ashfaq Ahmed.R[5]

[1]*(Dept. of Electronics and Instrumentation, Student of Adhiyamaan College of Engg. Anna Univ., Hosur,India)*
[2,3]*(Dept. of Information Science,Student of Auden Technology &Management Academy, VTU, Bangalore, India)*
[4]*(Dept. of Electronics and Instrumentation Engineering, Student of St.Peter's University. Chennai, India)*
[5]*(Dept of Computer Science, Ph.D Research scholar and Asst.Proffessor of Auden Technology & Management Academy,VTU, Bangalore, India)*

Abstract: *In this paper, we propose RegEx-Filter(pre-filtering approach) and IP traceback method to trace an unauthorized access incidents in the Internet, The current control technologies cannot stop specific way of access. The basic idea is to generate the RegEx print from RegEx set and use it in prefiltering most unmatched items and trace out the unauthorized user by means of traceback method. There are two key challenges in RegEx: the generation of RegEx print and the matching process of RegEx print. The generation of RegEx is tricky as it needs to tradeoff between two conflicting goals: filtering effectiveness, which means to filter out as many unmatched items as possible, and matching speed, which means that we want RegEx print to be as high as possible. Here we describe the development and the evaluation of our prototype system. The main features of our proposed method are the* ***RegEx-Filter*** *which filters the unmatched items.* ***Packet feature****, which is composed of specific packet information contained in a packet for identification of an unauthorized packet. And the* ***algorithm using datalink identifier*** *to identify a routing of a packet. We show the development of the prototype system, RegEx-Filter equipped with tracing functions on routers and its processing result.*

Keywords: *RegEx, Traceback model, Network security, Threat.*

I. Introduction

Regular expressions (RegEx) have been widely used in a variety of network and security applications. A Prefiltering Approach to Regular Expression Matching monitoring of network traffic based on application protocols. The RegEx matching problem can be defined as follows: given a set R of RegExes, at run time, for each incoming item i (*e.g.*, packets), we want to get RegEx set O(R, i) whose members are all matched by the item as a Deterministic Finite Automata (DFA) or Nondeterministic Finite Automata (NFA), prior RegEx matching solutions often fall into two categories: DFA-based and NFA-based. First, DFA-based solutions may achieve high speed because at any time there is only one active state, but may require too much memory[1]. For applications running on networking devices such as intrusion detection and prevention systems and application firewalls, RegEx matching needs to be done in high speed SRAM, which has small capacity in terms of a few megabytes. Second, NFA-based solutions require small memory, but cannot achieve high speed because at any time there may be many active states [4]. While the Internet as a business infrastructure increases its importance, the number of unauthorized access incidents on the Internet is growing, and such activity tends to cause a great problem. Nowadays installing Intrusion Detection Systems (IDS) coupled with firewalls, and monitoring networks enables us to quickly detect and react to unauthorized access shows a current dealing with unauthorized access. However, even if these tools can detect unauthorized activities, their sources cannot be identified. The reason is that denial of service (DoS) attacks, which have recently increased in number, can easily trace their sources of IP addresses using traceback method[2]. Thus, it is not possible for the access control alone to be a factor of unauthorized access. As the measure of unauthorized access, it is necessary to pinpoint the source in order to prevent the unauthorized activity. For this reason, we have been using a method to identify the source of an authorized activity and developed a prototype system.

II. Access With Permission

First, when user asks for permission to access the network, he needs to prove himself as authenticated user. This is done with help of identity-based RegEx mechanism[5]. This is done by entering the credentials such as ID and password, by user. Then on server side these credentials are matched with already stored there. If they matched the access to the network is granted otherwise denied. Identity-based is described in Figure1

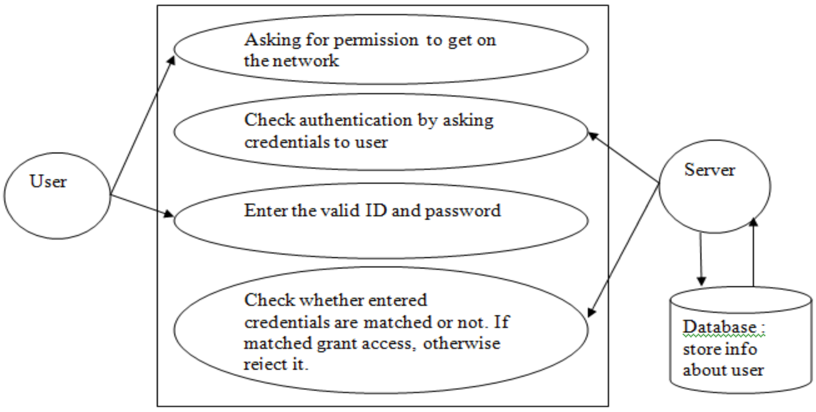
Asking for permission to get on the network
Check authentication by asking credentials to user
Enter the valid ID and password
Check whether entered credentials are matched or not. If matched grant access, otherwise reject it.
User
Server
Database : store info about user

Fig.1: Identity-Based Access

III. Regex Print Matching

As the RegEx print set in RegexFilter can be compiled into a composite DFA within limited memory, prior TCAM-based DFA matching solutions can be used here directly for high-speed RegEx print matching[6]. TCAM is a special type of memory which takes input of data as key to look-up address. It has the following three capacities: i) ternary states encoding: 0's, 1's, and *'s where *'s stand for either 0 or 1, enabling one TCAM entry to encode multiple DFA transitions; ii) parallel content lookup, enabling TCAM to complete lookups in a single operation no matter the number of occupied TCAM entries; iii) first- match semantic, making TCAM to return the index of the first address s *inside each state in the work is designed for TCAM-based packet classification applications, which produce prefix TCAM entries: the predicate of each entry is a prefix bit string (*e.g.*, 01**) where no 0 and 1 behind *. In fact, a ternary[7] TCAM entry allows * to appear at any positions (*e.g.*, 0**1), which means it misses the opportunity of encoding transitions created by non-prefix entries.

Table 1: Character Bonding

TCAM		SRAM
Current State	**Input System**	**Dest.State**
S_0	01000111	S_2
	01100111	S_2
	01000000	S_0
	01000000	S_0
	01000000	S_1
	01100000	S_1
	********	S_0

Table 2: Transistion Logic

TCAM		SRAM
Current State	**I/p System**	**Dest. State**
S_0	01*00111	S_2
	01*00**1	S_1
	01*00*1*	S_1
	01*001**	S_1
	********	S_0

Table 3: Optimal Encoding

TCAM		SRAM
Current State	**Input System**	**Dest. State**
S_0	01*00111	S_2
	*****000	S_0
	01*00***	S_1
	********	S_0

IV. Traceback Method

The ability required to perform traceback is "to identify the true IP address of the terminal originating attack packets." If we can identify the true IP address of the attacker's terminal, we can also get information about the organization (e.g. name or telephone number) involved [8] in the attack or the attacking terminal. As the method of the source pursuit of unauthorized access, some researches using IP (Internet Protocol) are performed. The source pursuit using IP is called IP Traceback.

IP traceback methods can be divided into two groups.

4.1. Proactive tracing:

This prepares information for tracing when packets are in transit. In a case where packet tracing is required, the target of the attack refers information and identifies the source of the packets.

4.2. Reactive Tracing:

This “reactive tracing” starts tracing when required. In our study, we have selected reactive tracing that does not increase network traffic at normal times and generates traffic for tracing only when actual tracing is required.

4.2.1 Hop-by-Hop Tracing

This method is to trace an IP packet from the target back to the source hop-by-hop, and trace the source based on the incoming packets that arrive one after another during a flood type attack.

4.2.2 Hop-by-Hop Tracing with Overlay Network

The particular problems involved in tracing routers hop-by-hop are that if there are too many hops, the number of necessary processing for tracing will be increased. As the result, it will take a longer time to trace, and information for tracing can be lost before trace processing is completed. Therefore, a method to build the overlay network for tracing purposes that involves a less number of hops is proposed. With this method, IP tunnels between the edge routers and the special tracking routers are created, and the IP packets are rerouted to the tracking router via IP tunnel. Hop-by-hop tracing is performed over the overlay network that consists of IP tunnels and tracking routers.

4.2.3 IPs Authentication

Another proposed technique is that when unauthorized access is detected, a Security association (SA) of the IPsec is created dynamically, and authenticating the packet with IPsec identifies the travel path and the source of the packet. Since this technique uses existing IPsec protocol, it has an advantage that it is not necessary to implement a new protocol.

4.2.4 Traffic Pattern Matching

Another proposed technique traces the forwarding path of the traffic by comparing traffic patterns observed at the entry and exit point of the network based on the network map.

V. Our Traceback Model

In this section, we describe our trace back architecture that identifies the source of a packet with forged source IP address. The architecture consists of the following three components:

5.1 Sensor

This component is deployed at target site has two functions. One is to detect unauthorized access from the network another is to request a manager to start tracing.

5.2 Tracer

This component implements a function in forwarding nodes to maintain information about forwarded IP packets as well as a function to trace the source of the forwarded packet along the attack path on forwarding unit.

5.3 Monitoring Manager

In response to a request from a sensor, this component controls tracers and manages the entire tracing process. We can install a tracer and a manager on each unit or install a single manager as a central manager of the entire network .

VI. Process Method

Basic model of our traceback method in practical terms, particularly network policy may restrict tracing a packet with certain limitation. We cannot trace a packet beyond our own network perimeter if neighboring networks impose different policy. We therefore adopt a ***distributed management*** approach that controls the tracing process and information within a particular group of networks. This control section is called as **A**utonomous **M**anagement **N**etwork (AMN) shown in fig.2. The monitoring manager, which is deployed in each AMN, executes a tracing process within its tracing process goes beyond the AMN’s boundary, the monitoring manager of the AMN that initiated the tracing process asks the monitoring manager in the adjacent AMN to trace the packet.

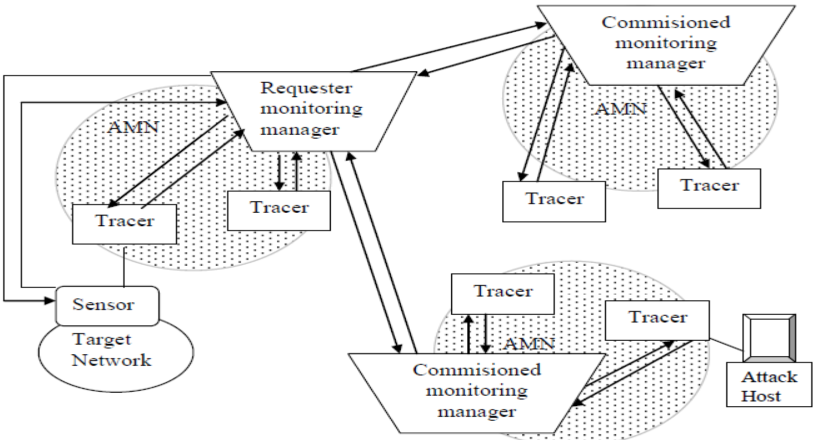

Fig.2: Autonomous Management Network

6.1 Process Flow:

Our traceback approach involves several Steps, from attack detection to source identification, Sensors are deployed at each target network. When a sensor detects an attack, it creates data containing features of the attack packet and sends a tracing request to the monitoring manager deployed in its AMN[2]. The monitoring manager order the AMN's tracer to trace the attack packet. The tracer identifies the adjacent node and returns the result to the monitoring manager. Based on the result returned, the process described above continues until the tracer identifies the attack packet's source.If a tracing process goes beyond the AMN's boundary, processing is handed over to the relevant monitoring manager that controls that AMN. The monitoring managers in each AMN traces the packet in their AMN and sends the tracing result to the monitoring manager that initiated the traceback request. The requester monitoring manager sends the final results to the sensor that requested the trace.

6.2 Trace Algorithm

We have developed the algorithm that processes Trace Order reception, trace execution for upstream path decision and trace report. Below we describe our algorithm.

Step 1: Start the Tracing process.
Step 2: Receive the packet feature and passed it to packet search module.
Step 3: Check Packet Information Area with packet feature received.
Step 4: If any match found then let matched record as target record.
Step 5: Compares the Address Information (i.e MAC address) in target record with the address Information (i.e MAC address &IP address)of the connected tracer stored in trace information.
Step 6: If match found decide IP address and return to the monitoring manager as trace result.
Step 7: Repeat step 3 to step 6 until source of the attack is detected.
Step 8: Stop the Tracing Process.

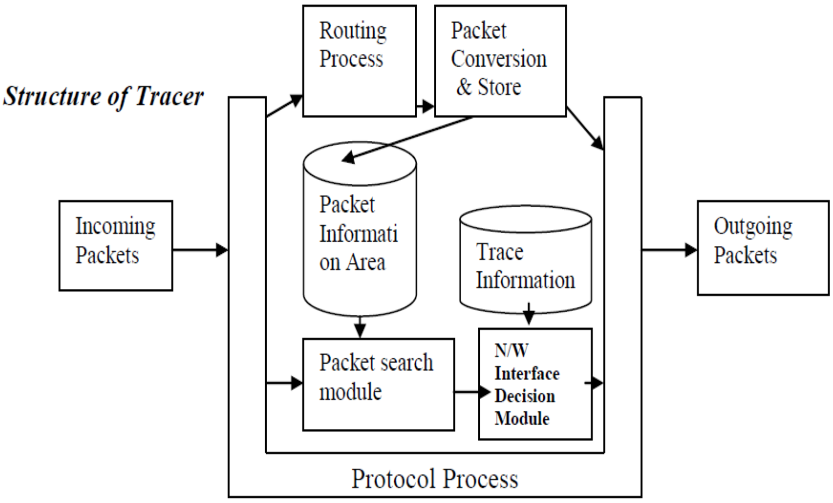

Fig.3: Structure of Tracer

VII. Proposed Architecture

In this architecture, the technique of tracing the unauthorized access of data using RegEx print and Traceback method as the current control technologies cannot stop specific way of access. As shown in the figure the Regular Expression filter, filters all the unmatched items which accessed by the unauthorized user. Once the unmatched items are filtered by the RegEx, then it traces the protocol of the unauthorized user. After getting the IP address of the user, it sends to the administrator to control the action on the threat. Then the action continous by sending the data to information management where information is treated in two ways: packet information, contains packet features which includes network interface information and forwarding time of the packet necessary information for tracing. and information tracer, traces the information using Trace table method, ARP table method and Order driven query method.

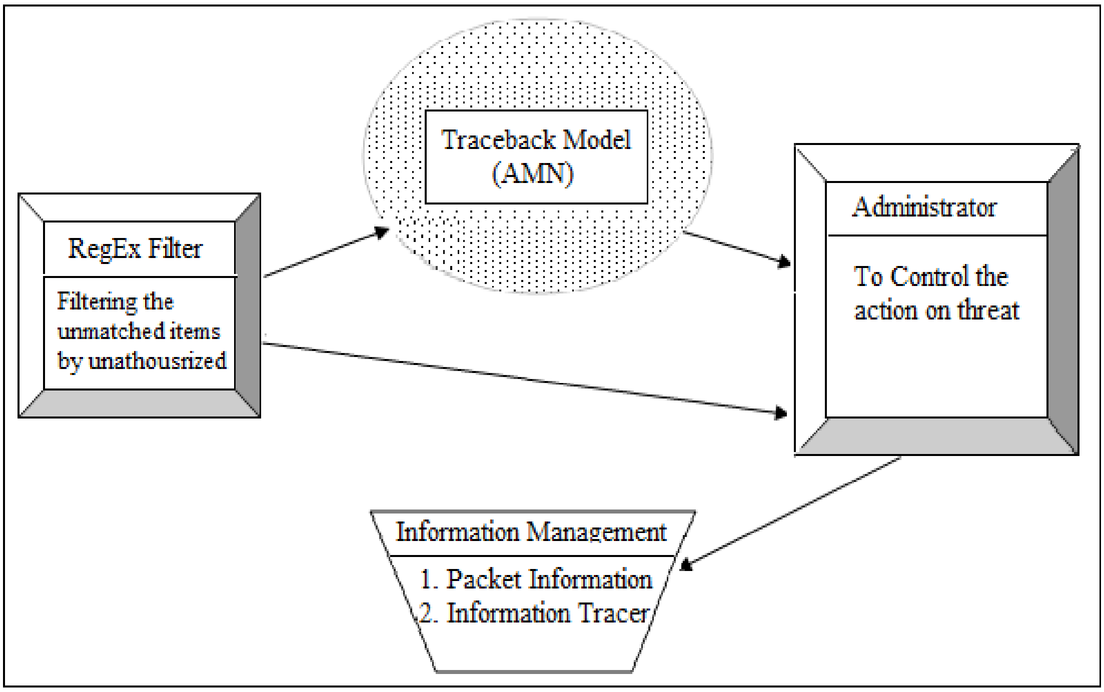

Fig.4: Proposed Architecture

7.1 Information Management

There are two types of information used in tracers. One is the packet information that converts traversed packets information into packet features and stores them, and the other is the network interface information that stores network interface information between two units connected each other.

7.1.1 Packet Information Area

Packet Information Area contains packet features which includes network interface information and forwarding time of the packet necessary information for tracing. On our implementation, records are stored in the memory area of the tracer for the purpose of real-time processing. If the volume of Packet Information Area exceeds the memory capacity, the oldest record will be deleted and the latest one will be stored in turn.

7.1.2 Trace Information

We have studied three methods for obtaining network interface information from the unit connected with the tracer.

Method 1: Trace table method

Checking the network interface number, IP addresses and physical addresses (e.g. MAC address on LAN) of the connected tracers in advance, and storing them in the unit.

Method 2: ARP table method

Using the ARP table stored in the unit to look up the IP address and physical address of the connected tracer when Trace Order is received.

Method 3: Order-driven query method

Without providing a fixed table, obtaining network interface information using the lower layer protocols (e.g. RARP protocol) in response to Trace Order. We have reviewed each method and reached the following conclusion: As network interface information is temporarily stored in the ARP table, some information may be changed when searching the table; Although the order-driven query method is suitable for obtaining the latest network interface information, the process is complicated and takes longer time because the query task to the adjacent node is called every time a trace order is issued. Therefore, we select the trace table method that provides real-time, reliable, and efficient tracing.

VIII. Conclusion

In this paper, we present the filtering method where the data used by the user for which the permissions are not provided are being eliminated using using RegEx-Filter. And also we proposed the tracing method which traces the IP address of unauthorized user. The possiblily of implementing the tracer function on all the network equipments in a network environment which not secure. By using these two techniques we ensure that a data in the network will be protected from the hackers by providing a secure environment.

Reference

[1] Lichuin Bao (2008) “Location Authentication Methods for Wireless Network Access Control”, International conference on computer engineering and technology

[2] Yabin Liu, Huanguo Zhang , Huanguo Zhang and Bo Zhao(2009) Research on Unified Network Access Control Architecture 2009 IEEE Ninth International Conference on Computer and Information Technology. Regular Expression Matching for Deep Packet Inspection. In: Proc. ACM/IEEE ANCS. (2006) 93–102

[3] Ramaswamy, R., Kencl, L., Iannaccone, G.: Approximate Fingerprinting to Accelerate Pattern Matching. In: Proceedings of the 6th ACM SIGCOMM conference on Internet measurement. (2006) 301–306.

[5]. Liu, T., Yang, Y., Liu, Y., Sun, Y., Guo, L.: An Efficient Regular Expressions Compression Algorithm From A New Perspective. In: Proc. IEEE INFOCOM. (2011) 2129–2137

[4] Broder, A., Mitzenmacher, M., Mitzenmacher, A.: Network Applications of Bloom Filters: A Survey. In: Internet Mathematics, Citeseer (2002)

[6]. Kumar, S., Dharmapurikar, S., Yu, F., Crowley, P., Turner, J.: Algorithms to Accelerate Multiple Regular Expressions Matching for Deep Packet Inspection. In: Proc. ACM SIGCOMM. (2006) 339–350

[7]. Ficara, D., Giordano, S., Procissi, G., Vitucci, F., Antichi, G., Di Pietro, A.: An Improved DFA for Fast Regular Expression Matching. ACM SIGCOMM Computer Communication Review 38(5) (2008) 29–40