Cyber Security: Efficient Searching Algorithm for Intrusion Detection

Siksha 'O' Anusandhan
Deemed to be University
Bhubaneswar, Odisha, India

Intrusion Detection System

Intrusion Detection System

Intrusion Detection Systems (IDSs) are one of the most useful tools to identifying malicious attempts over the network and protecting the systems without modifying the end-user software.

- Firewalls only check specified fields of the packet headers but IDS checks the body of the packets.
- IDSs detect the malicious information in the payloads.
- An IDS typically contains a database that describes the signatures of malicious behavior.
- The number of patterns is generally a few thousands and still increasing.

IDS Algorithm

Algorithm

An algorithm is a procedure used for solving a problem or performing a computation.

-One of the important algorithm in IDS is **String Matching Algorithm**.

Objective of the algorithms: From a stream of packets, the algorithm identifies those packets that contain data matching the signatures of a known attack.

IDS Algorithm

Algorithm

An algorithm is a procedure used for solving a problem or performing a computation.

-One of the important algorithm in IDS is **String Matching Algorithm**.

Objective of the algorithms: From a stream of packets, the algorithm identifies those packets that contain data matching the signatures of a known attack.

We can classified a packet using **single pattern string matching** algorithm or **multiple pattern string matching** algorithm.

IDS Algorithm

Single pattern string matching

In single pattern string matching the packet is searched for a single string at a time.

Multiple pattern string matching

In multiple pattern string matching searches the packet for the set of strings all at once.

Today's class we will only consider single pattern string matching algorithm.

IDS Algorithm

String Matching Algorithm:

Step 1: Read the packet.

Step 2: Use **Strstr function** which searches a given string in the database and returns the position (address) pointer to string. If string is found then reture the index of first position otherwise return 0 if it is not found.

Step 3: Use **Strchr function** searches a character in the database. The function returns the position (address) pointer to a character, where the character was first found, or 0 if not found.

IDS Algorithm

Pattern Matching Algorithm:

The Pattern Matching Algorithm can be divided into two phases- **preprocessing phase** and **search phase.**

Preprocessing phase:

- The first task of preprocessing phase is to change each byte from signature string to two bytes and put these bytes in converting array.
- The second task is to generate a two dimensional array called NEXT.
- This Array is very important which decides how to move to a proper position in the next search.
- After, array NEXT is generated, its values will be invariable during searching process.

Searching phase:

- The comparison is performing from right to left at each check point.
- If a mismatching occurs, the next to the last character of current comparing window is used to execute the next matching.

Types of network attacks

Attack Name	Description	Attack by (Packets, Tools, etc.)
Jamming Attack	By using the channel that they are communicating on, it prohibits other nodes from accessing it to connect.	Radio frequency noise.
Flooding	A DoS attack in which a server receives many connection requests but does not reply to complete the handshake. (ICMP Flood, SYN Flood, HTTP Flood).	Unbound number of requests without acknowledgment of packet after receiving it.
Smurf Attack	A network layer DDoS attack caused due to the network tools misconfiguration.	Source IP fooling victim IP.
Ransomware	A form of malware that infiltrates and encrypts important files and systems, preventing a person from accessing their own data.	B0r0nt0k (encryption ransomware), Mado (malicious program)
Session Hijacking	To obtain unauthorized access to the Web Server, the Session Hijacking attack disrupts the session token by stealing or guessing a valid session token (e.g., predictable session token)	Malicious JavaScript Codes, XSS, Session Sniffing.

Class Assignment

Develop a signature-based intrusion detection system (IDS) with a dictionary file (malicious.txt) that stores all malicious information [IP address, port number]. Now suppose the system captures incoming packets (data.txt) and detects if the file is malicious. If the IP address or port number matches the dictionary date, the malicious file is returned, otherwise, the packet is accepted.

Class Assignment

Develop a signature-based intrusion detection system (IDS) with a dictionary file (malicious.txt) that stores all malicious information [IP address, port number]. Now suppose the system captures incoming packets (data.txt) and detects if the file is malicious. If the IP address or port number matches the dictionary date, the malicious file is returned, otherwise, the packet is accepted.

Sample Input

Read the dictionary and store in an array.
Read incoming packets and compare the IP address and port number with the dictionary array.

Sample output

The incoming file is a malicious packet **or** incoming packet is accepted.

Class Assignment

Packet **malicious.txt**

IP address	Port number
119.115.103.96	2001/tcp
103.126.161.114	2140/udp
117.82.77.153	2989/tcp
162.247.74.74	12346/tcp
171.25.193.77	20433/udp
54.37.203.143	4950/tcp
192.142.133.7	5390(DNS)
58.215.218.170	8443 (HTTP)

Class Assignment

Packet **data.txt**

Frame 13: 55 bytes on wire (440 bits), 55 bytes captured (440 bits) on the interface.

Source MAC: AzureWav$_6c : bc : e5(dc : f5 : 05 : 6c : bc : e5)$

Destination MAC: ca:a5:be:7e:a2:22 (ca:a5:be:7e:a2:22)

Internet Protocol Version 4

Source IP: 117.82.77.153

Destination IP: 142.250.193.138

Transmission Control Protocol, Src Port: 5390, Dst Port: 443, Seq: 1, Ack: 1, Len: 1

Source Port: 5390

Destination Port: 443

TCP payload (1 byte)