




INTRUSION DETECTION






Intrusion and Intrusion Detection

- Intrusion : Attempting to break into or misuse your system.
 - Intruders may be from outside the network or legitimate users of the network.
 - Intrusion can be a physical, system or remote intrusion.
- 




Different ways to intrude

- Buffer overflows
 - Unexpected combinations
 - Unhandled input
 - Race conditions
- 



Intrusion Detection Systems (IDS)

Intrusion Detection Systems look for attack signatures, which are specific patterns that usually indicate malicious or suspicious intent.






Intrusion Detection Systems (IDS)


- Different ways of classifying an IDS


IDS based on

- anomaly detection
 - signature based misuse
 - host based
 - network based
- 




Anomaly based IDS

- This IDS models the normal usage of the network as a noise characterization.
 - Anything distinct from the noise is assumed to be an intrusion activity.
 - E.g flooding a host with lots of packet.
 - The primary strength is its ability to recognize novel attacks.
- 




Drawbacks of Anomaly detection IDS

- Assumes that intrusions will be accompanied by manifestations that are sufficiently unusual so as to permit detection.
 - These generate many false alarms and hence compromise the effectiveness of the IDS.
- 



Signature based IDS


- This IDS possess an attacked description that can be matched to sensed attack manifestations.
 - The question of what information is relevant to an IDS depends upon what it is trying to detect.
 - E.g DNS, FTP etc.
- 

Signature based IDS (contd.)

- ID system is programmed to interpret a certain series of packets, or a certain piece of data contained in those packets, as an attack. For example, an IDS that watches web servers might be programmed to look for the string "phf" as an indicator of a CGI program attack.
- Most signature analysis systems are based off of simple pattern matching algorithms. In most cases, the IDS simply looks for a sub string within a stream of data carried by network packets. When it finds this sub string (for example, the "phf" in "GET /cgi-bin/phf?"), it identifies those network packets as vehicles of an attack.





Drawbacks of Signature based IDS

- They are unable to detect novel attacks.
 - Suffer from false alarms
 - Have to programmed again for every new pattern to be detected.
- 





Host/Applications based IDS

- The host operating system or the application logs in the audit information.
 - These audit information includes events like the use of identification and authentication mechanisms (logins etc.) , file opens and program executions, admin activities etc.
 - This audit is then analyzed to detect trails of intrusion.
- 




Drawbacks of the host based IDS

- The kind of information needed to be logged in is a matter of experience.
 - Unselective logging of messages may greatly increase the audit and analysis burdens.
 - Selective logging runs the risk that attack manifestations could be missed.
- 




Strengths of the host based IDS

- Attack verification
 - System specific activity
 - Encrypted and switch environments
 - Monitoring key components
 - Near Real-Time detection and response.
 - No additional hardware
- 




Stack based IDS

- They are integrated closely with the TCP/IP stack, allowing packets to be watched as they traverse their way up the OSI layers.
 - This allows the IDS to pull the packets from the stack before the OS or the application have a chance to process the packets.
- 




Network based IDS

- This IDS looks for attack signatures in network traffic via a promiscuous interface.
 - A filter is usually applied to determine which traffic will be discarded or passed on to an attack recognition module. This helps to filter out known un-malicious traffic.
- 




Strengths of Network based IDS

- Cost of ownership reduced
 - Packet analysis
 - Evidence removal
 - Real time detection and response
 - Malicious intent detection
 - Complement and verification
 - Operating system independence
- 




Commercial ID Systems

- ISS – Real Secure from Internet Security Systems:
 - Real time IDS.
 - Contains both host and network based IDS.
 - Tripwire – File integrity assessment tool.
 - Bro and Snort – open source public-domain system.
- 




Bro: Real time IDS

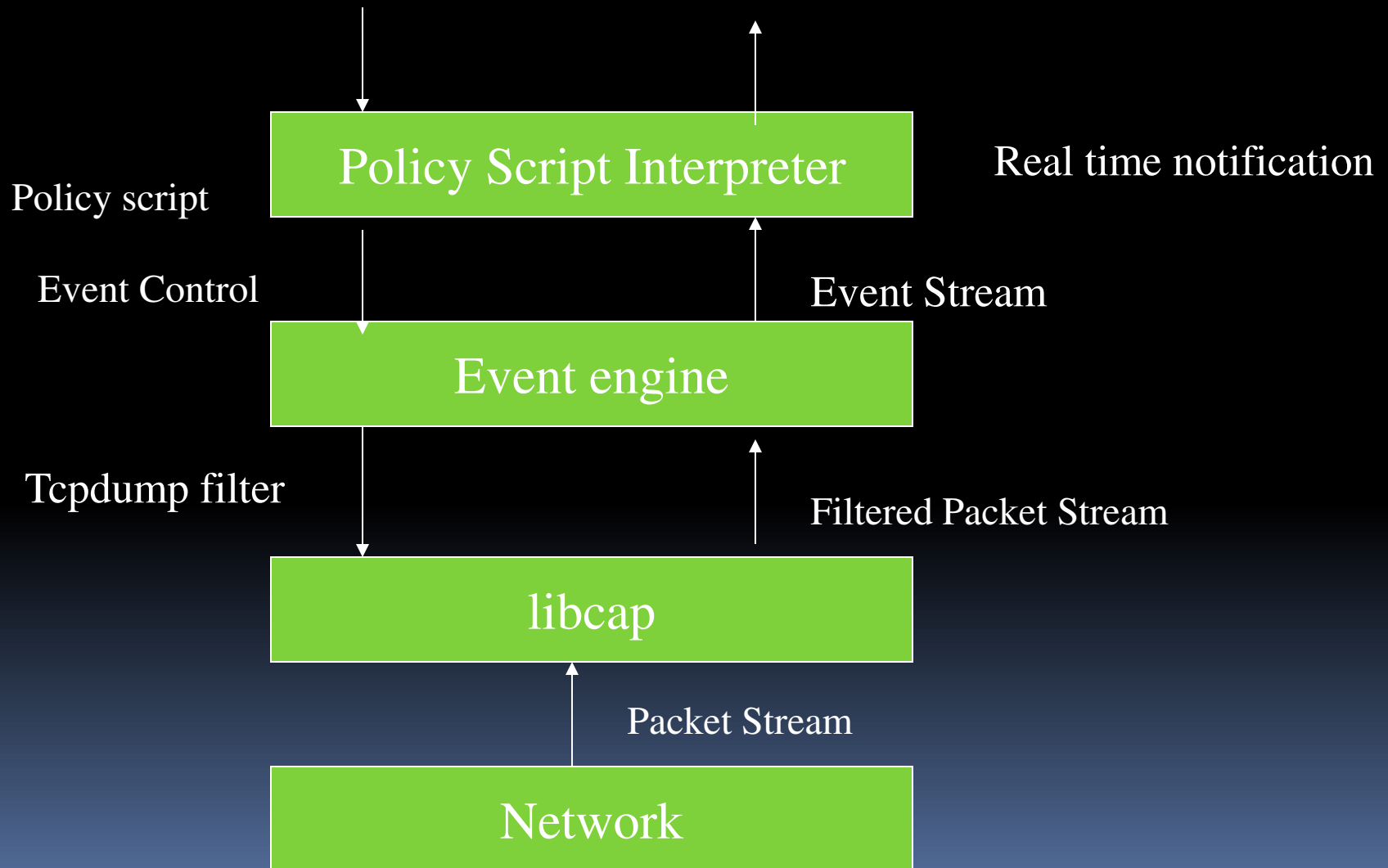
- Network based IDS
 - Currently developed for six Internet applications: FTP, Finger, Portmapper, Ident, Telnet and Rlogin.
- 



Design goals for Bro

- High-speed, large volume monitoring
 - No packet filter drops
 - Real time notification
 - Mechanism separate from policy
 - Extensible
 - Monitor will be attacked
- 

Structure of the Bro System



Bro - libcap


- It's the packet capture library used by tcpdump.
- Isolates Bro from details of the network link technology.
- Filters the incoming packet stream from the network to extract the required packets.
- E.g port finger, port ftp, tcp port 113 (Ident), port telnet, port login, port 111 (Portmapper).
- Can also capture packets with the SYN, FIN, or RST Control bits set.

Bro – Event Engine

- The filtered packet stream from the libcap is handed over to the Event Engine.
- Performs several integrity checks to assure that the packet headers are well formed.
- It looks up the connection state associated with the tuple of the two IP addresses and the two TCP or UDP port numbers.
- It then dispatches the packet to a handler for the corresponding connection.




Bro – TCP Handler

- For each TCP packet, the connection handler verifies that the entire TCP Header is present and validates the TCP checksum.
 - If successful, it then tests whether the TCP header includes any of the SYN/FIN/RST control flags and adjusts the connection's state accordingly.
 - Different changes in the connection's state generate different events.
- 



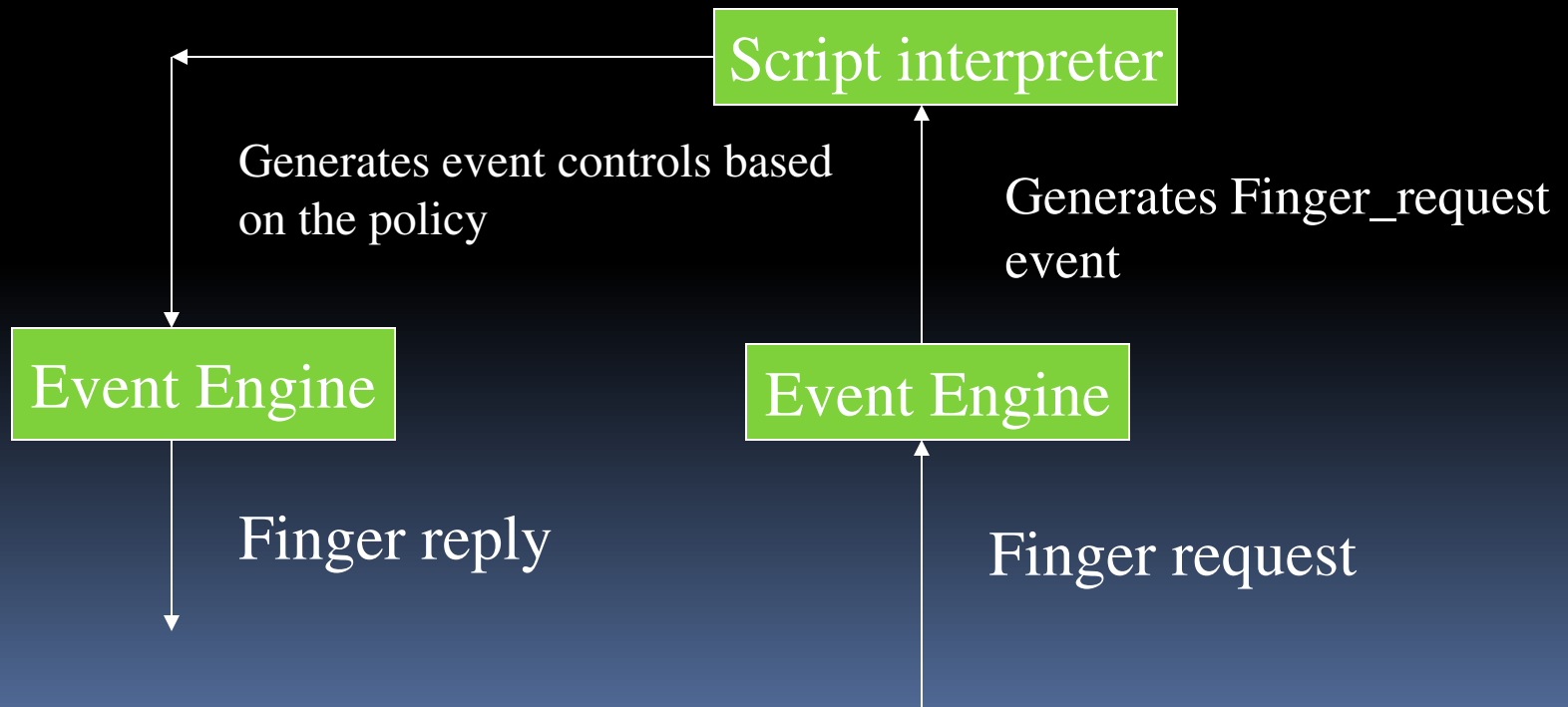
Policy Script Interpreter

- The policy script interpreter receives the events generated by the Event Engine.
 - It then executes scripts written in the Bro language which generates events like logging real-time notifications, recording data to disk or modifying internal state.
 - Adding new functionality to Bro consists of adding a new protocol analyzer to the event engine and then writing new events handlers in the interpreter.
- 

Application Specific Processing

- Finger

Tests for buffer overflow,
checks the user against
sensitive ids, etc





Future of IDS

- Usage of Artificial Intelligence
 - Use of Neural Networks
 - Use of Genetic Algorithms.
- 