Compiling Natural Language Expressions to Extended BPF Programs for Stateful Network Policy Enforcement

Mohammad Firas Sada , Nik Sultana

ILLINOIS TECH
College of Computing

Introduction

• The GAPS CLOSURE project, funded by the government, is developing a natural language framework for cross-domain data filtering and transformation: https://github.com/gaps-closure

• This allows for expressing packet filtering and transformation rules in plain English, no specialized knowledge or reprogramming for different systems/hardware needed.

Motivation

• Network security is crucial for national security and is increasing in importance.

• Firewall rules protect against incoming malware, DoS attacks, ransomware, and outbound traffic.

• Translating English to packet/network-level metrics reduces misconfiguration and enables richer configurations.

• Stateful rules are crucial for network security, identifying handshakes, parts of TCP flows, and 5-tuples for connection tracking.

Approach

• Simplify network/system configuration using plain English in NRL to produce BPF instructions for packet filtering, transformation, and network statistics constraints.

• NRL is a restricted English language for expressing constraints and transformations over models, while BPF is a 3-decade-old instruction set for packet filtering.

• NRL rules apply to any data format defined by a DFDL Schema.

NRL Rules + Data Format = BPF Bytecode

• Extract packet field sizes and offsets from DFDL schema to generate BPF bytecode for filtering, transforming, and network stats.

• Chicago BPF enables stateful packet filtering and transformation using scratch memory, including storing packet 5-tuples for complex packet filtering and transformation rules.

• Stateful rules like SYN/ACK handshake identification and targeting TCP flow packets can be implemented using scratch memory.

Acknowledgements

The team members at **Peraton Labs** for their feedback and input.

This work is supported by the **Defense Advanced Research Projects Agency (DARPA)** under the GAPS program. The views, opinions and/or findings in this poster should not be interpreted as representing the official views or policies of the Department of Defense or the US government.

Results

• A transpiler written in Java, that taken NRL as input, and takes the definition of a packet file data format.
• It outputs equivalent BPF bytecode that runs on the hardware target to filter and transform packet data according to the NRL rules.

• The transpiler can take in complex policies composed of hundred of rules, and instantly translate the rules into one BPF policy.

NRL Input

```
* If Protocol is ‘6’ then
IPSrc must be ‘192.168.1.1’
* If PortSRC is ‘1234’ then
TTL > 0
* If Protocol is ‘17’ set
PortSRC to ‘8888’
* If packet is even set IPDest
to ‘192.168.1.1’
•  If IPSrc = ‘192.168.1.1’
   then set Memory.M1 to
   IPDest;
```

Transpiler

BPF Output

```
(000)   ldh [12]
(001)   jeq #0x86dd jt 2 jf 6
(002)   ldb [20]
(003)   jeq #0x6 jt 4 jf 15
(004)   ldh [54]
(005)   jeq #0x16 jt 14 jf 15
.
.
.
```

10TH ANNUAL HOT TOPICS in the SCIENCE OF SECURITY
APRIL 3 - 5, 2023 | Virtually hosted by The National Security Agency
hotsos.org