Washington University in St. Louis

A Framework for Rule Processing in Reconfigurable Network Systems

Michael Attig and John Lockwood

Washington University in Saint Louis
Applied Research Laboratory
Department of Computer Science and Engineering
May 1, 2005

Outline

- Overview
- Background
- Architecture
- Results
- Summary

Rule Processing Overview

- Rule processing & intrusion detection

- TCP Flow Processing
- Header Processing
- Payload Scanning

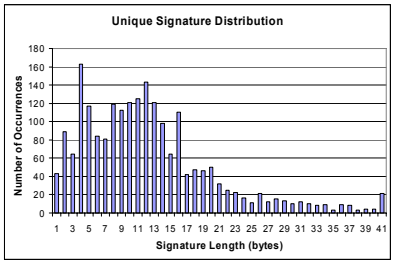
alert tcp any 110 → any any (msg:"Virus - Possible MyRomeo Worm"; flow:established; content:"myromeo.exe"; nocase; classtype:misc-activity; sid:723; rev:6;)

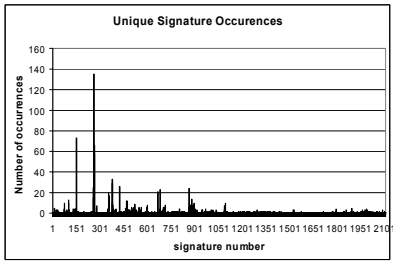
- Snort Rules (version 2.2 Sept 2004)
  - 2464 Rules
  - 292 Headers
  - 2107 Signatures
  - 233 Regular Expressions

Rule Characteristics

- 2464 unique rules
- 292 unique header rules
  - 168 are "header-only"
- 2107 unique signatures
  - 97% less than 32 bytes
  - Spread across 2296 of rules
- 233 regular expressions
  - Snort rules always contain static signature also

- Few signatures associated with many rules
  - 83% found in single rule
  - Only 18 associated with more than 10 rules
- 10 header rules can match at once (pessimistic)





Snort version 2.2 (Sept 2004)

Fully Functional Rule Processing

String Matching

? Comparators [Sourdis fccm'04]

Partitioning [Baker fccm'04]

? TCAMs [Yu hoti'04]

? ?

Bloom Filters [Attig fccm'04]

DFAs [Moscola fccm'03] ?

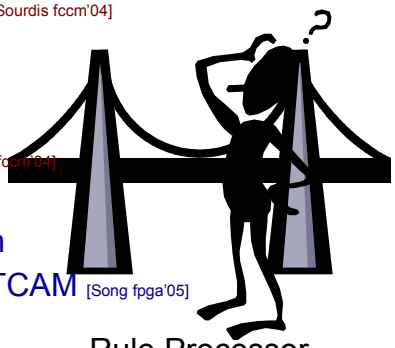
Header Classification

NFAs [Clark fccm'04] BV-TCAM [Song fpga'05]

? TCP Flow ?

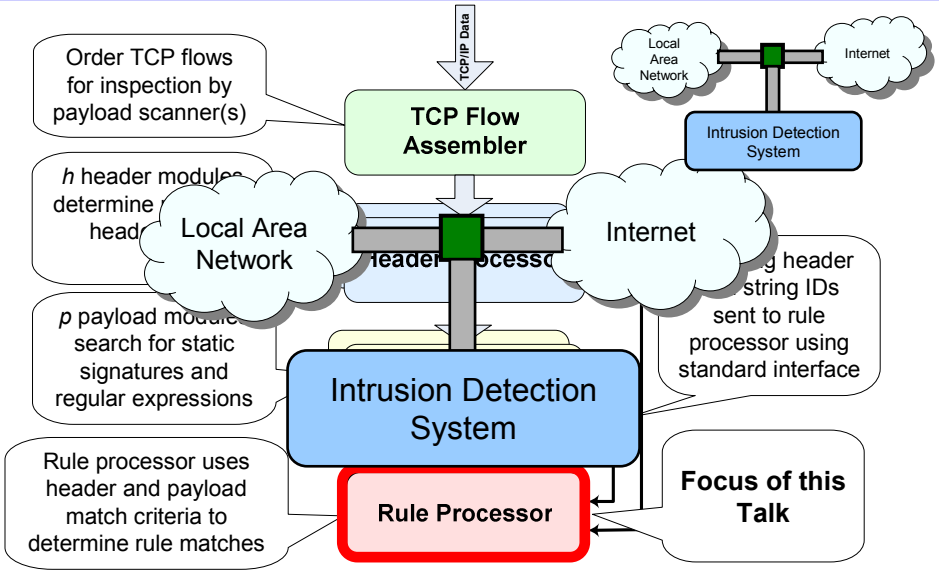
Reconstruction [Schuehler fpl'04]

Pipelining [Cho fccm'04]

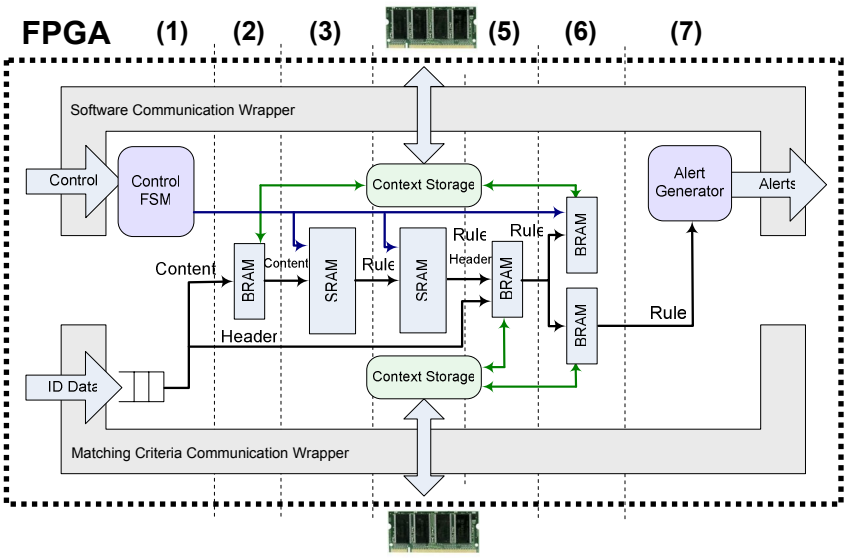


Rule Processor

Rule Processing Framework Overview



Rule Processor

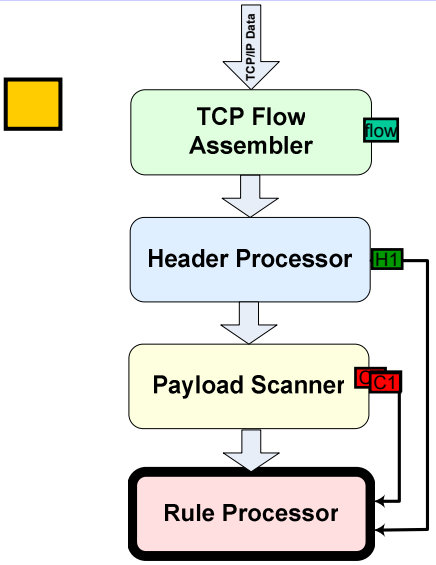


Example

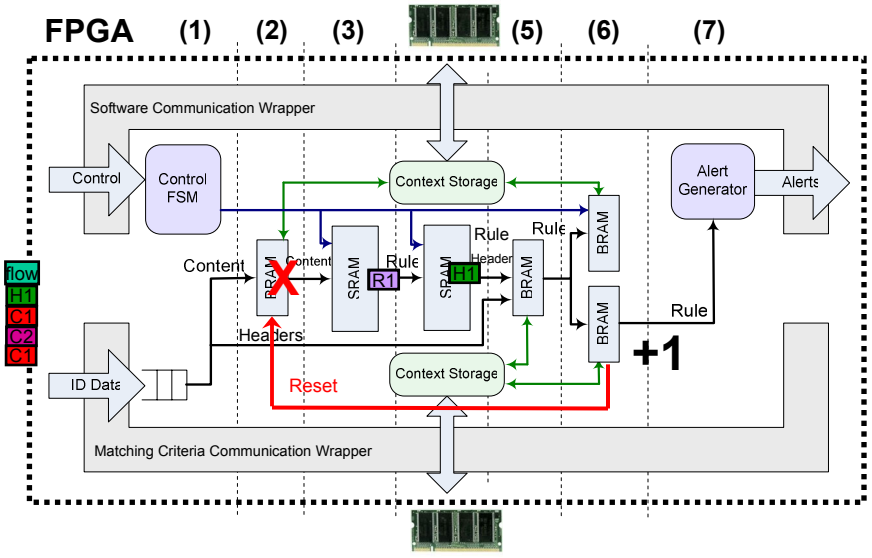
R1: Alert tcp any 80 → any 125
(content:“string1”; content:“string2”;)

Algorithmically:
R1: H1 Λ C1 Λ C2

Rule Processing Example

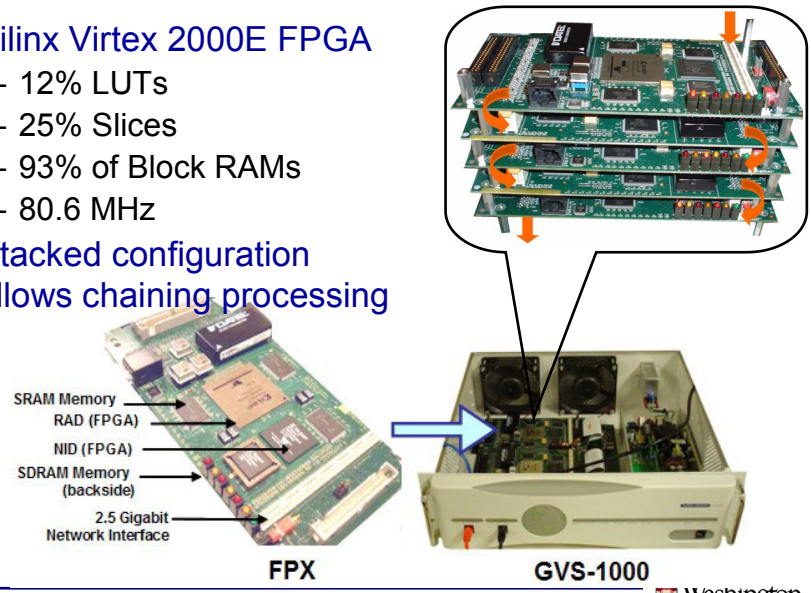


Rule Processor



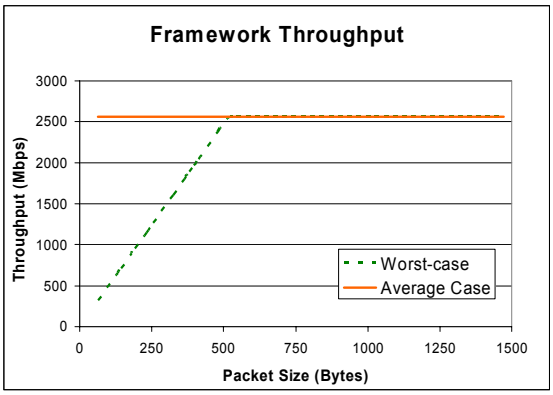
Implementation Environment

- Xilinx Virtex 2000E FPGA
  - 12% LUTs
  - 25% Slices
  - 93% of Block RAMs
  - 80.6 MHz
- Stacked configuration allows chaining processing

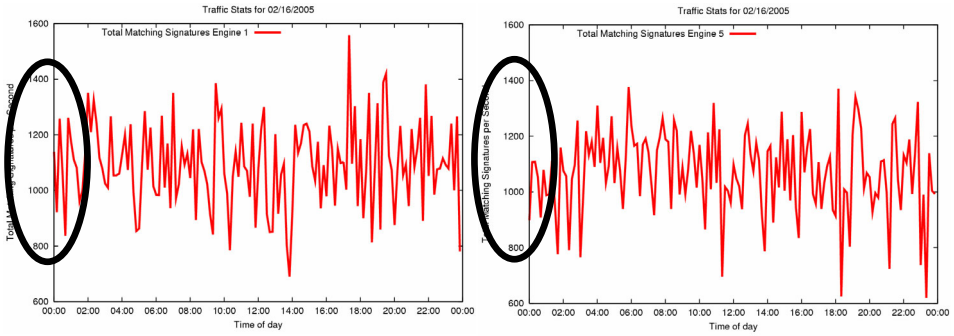


Worst Case Throughput

- Worst case when signature associated with many rules detected
- Only 18 signatures in more than 10 rules
- Worst case signature
  - |00 00 00 00| in 135 rules
- Scenario:
  - Back-to-back 44 byte TCP packets from different flows ***and*** |00 00 00 00| as payload
  - Worst case assumes 7 million attack packets per second



Intrusion Detection of WashU's Backbone Network



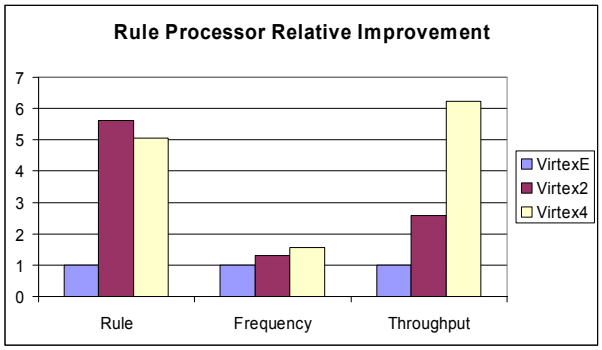
Matching 4 Byte Signatures

Matching 12+ Byte Signatures

- Observe ~10,000 total string matches per second on WashU's backbone network (~250-300 Mbps)
- Scaling to 2.5 Gbps, only ~100,000 string matches per second

Next Generation FPGA Projections

- More block RAM
- Faster place & route
- Parallel copies of pipeline
  - Multiple IDs per clock cycle
- QDR SRAMs
- 6x improvement to throughput



Related Work

Function	Group and Component	Device	Logic Cells	Throughput (Gbps)
Flow Monitoring	GaTech Stream Assembler	Virtex 1000	876 (10%)	3.2
	Northwestern U. Flow Monitor	Virtex2-8000	-	48.3
	WashU TCP Processor	Virtex4 140	22,100 (35%)	10.3
Header Processing	WashU BV-TCAM	Virtex4 100	4,200 (10%)	10
Payload Scanning	Crete Pre-decoded CAMs	Virtex2-6000	64,268 (95%)	9.7
	GaTech Decoder Trees	Virtex2-8000	54,890 (81%)	7
	Tokyo Trie-based Hash	Virtex2-6000	2,365 (7%)	10
	UCLA Packet Filters	Spartan 3 2000	15,202 (37%)	3.2
	USC Partitioning	Virtex2 Pro	15,010 (15%)	4.5
	WashU Bloom Filters	Virtex4 100	35,850 (85%)	20.4
Correlation	WashU Rule Processor	Virtex4 100	40,200 (95%)	15.9

Contributions

- Development of large-scale Rule Processing Framework
  - Bridge between component processing and rule processing
  - Supports up to 32,768 rules
- Rule processing framework capable of 2.5 Gbps throughput on FPX
  - Projected to 15.9 on latest Virtex 4
- Rule processor operated on TCP flows
  - Context information stored for over 2 million simultaneous flows

Acknowledgments

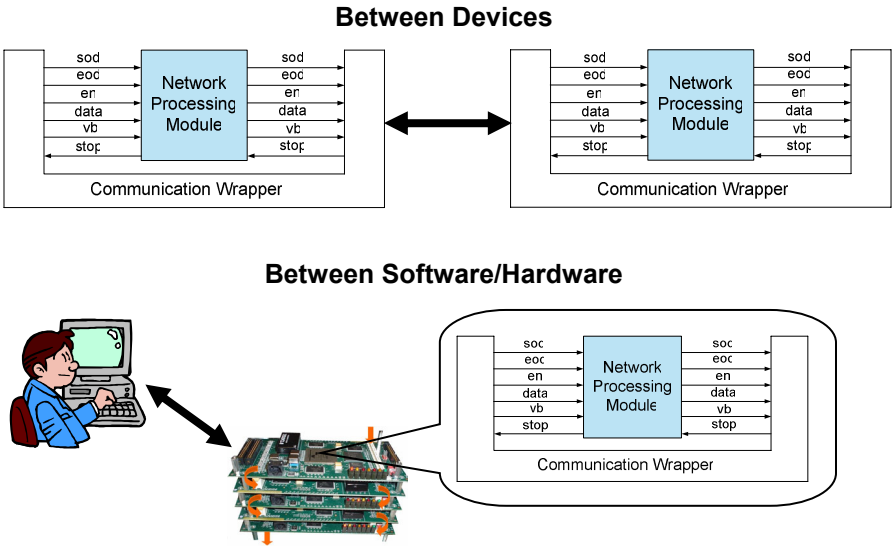
- Research Sponsors
  - Global Velocity
  - Boeing

- ARL Faculty & Students

http://arl.wustl.edu/projects/fpx/reconfig.htm

Questions?

Communication Wrapper Interface

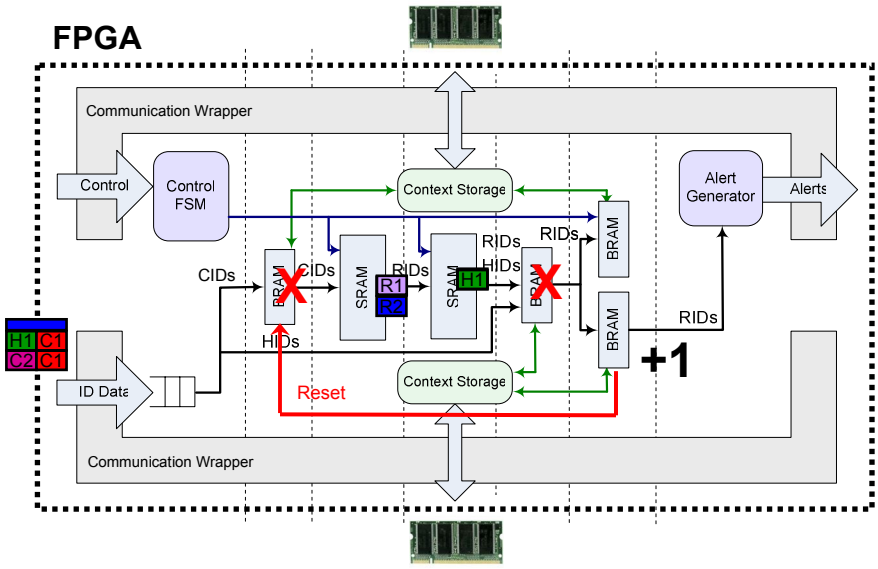


Example

R1: Alert tcp any 80 → any 125
(content:“string1”; content:“string2”;)
R2: Alert tcp any 8080 → any 1024
(content:“string1”;)

R1: H1 Λ C1 Λ C2
R2: H2 Λ C1

Rule Processor



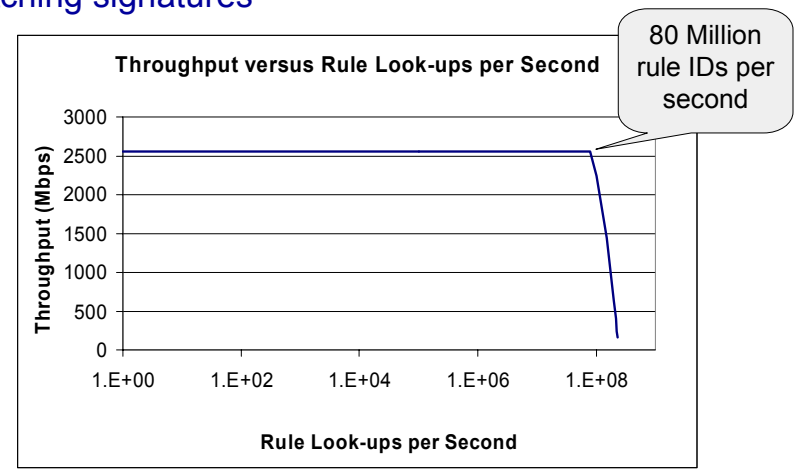
Adding Modules

- Accept and act upon IP packets
- Communicate match criteria using communication wrapper interface
  - Provide deterministic interfaces
  - Abstract transport protocol
- Software Configuration using communication wrapper
- Represent matching criteria as ID numbers

- Allows combination of techniques
  - Take advantage of best characteristics
    - General classifiers vs. field-specific headers
    - Static strings vs. regular expressions

Evaluation

- Recall Rule IDs are inserted into pipeline based on matching signatures



Additional Rules Supported

- Virtex 2
  - 120 of 144 Block RAMs (18 Kbits each)
  - 2 copies of pipeline
    - 10 BRAM in stage 2
    - 10 BRAM in stage 5
    - 40 BRAM in stage 6
  - 184,320 rules supported
- Virtex 4
  - 216 of 240 Block RAMs (18 Kbits each)
  - 4 copies of pipeline
    - 9 BRAM in stage 2
    - 9 BRAM in stage 5
    - 36 BRAM in stage 6
  - 165,888 rules supported