

A NAIVE IMPLEMENTATION OF BLINDBOX: PROTOCOL I

Deep Packet Inspection over Encrypted Traffic

Seyma Bodur, Fatma Demirtas, Cansin Yildiz

OUTLINE

- Introduction and Motivation
- BlindBox: Deep Packet Inspection over Encrypted Traffic
 - System Overview
 - Threat Model
 - Evaluation Highlights
- A Naive Implementation of BlindBox: Protocol I
 - System Overview
 - Demo
 - Limitations
- Questions?

INTRODUCTION AND MOTIVATION

WHAT IS DEEP PACKET INSPECTION (DPI)?

- In-network middleboxes use DPI to examine and alter packets
- Used to enforce security policies
 - Intrusion detection/prevention, exfiltration prevention, parental filtering etc.

DPI AND HTTPS

- HTTPS and other encryption protocols have dramatically grown in usage
- Packet payloads are encrypted, middleboxes can no longer inspect them
- To enable inspection, some systems support *insecure* HTTPS
 - Man-in-the-middle attack on SSL

Functionality of
Middleboxes

or

Privacy from
Encryption

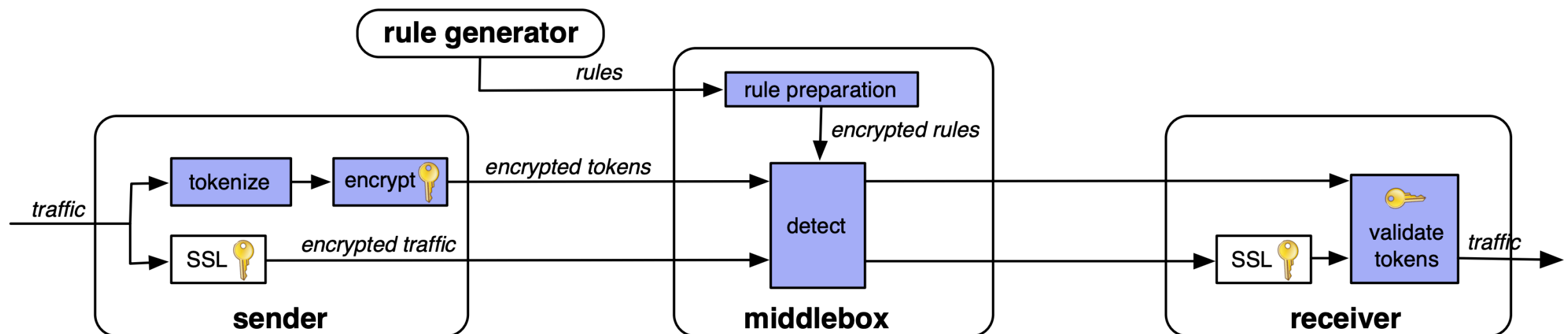
Can we get both?

BLINDBOX: DEEP PACKET INSPECTION OVER ENCRYPTED TRAFFIC

Justine Sherry, Chang Lan, Raluca Ada Popa, Sylvia Ratnasamy

BLINDBOX: BOTH PRIVACY AND DPI

- Detection
 - Middlebox receives both SSL-encrypted traffic and encrypted tokens
 - Detect module searches for matches between encrypted rules and encrypted tokens
- Receive
 - Receiver decrypts and authenticates traffic using normal SSL
 - Receiver also checks that encrypted tokens were encrypted properly by sender



THREAT MODEL SUMMARY

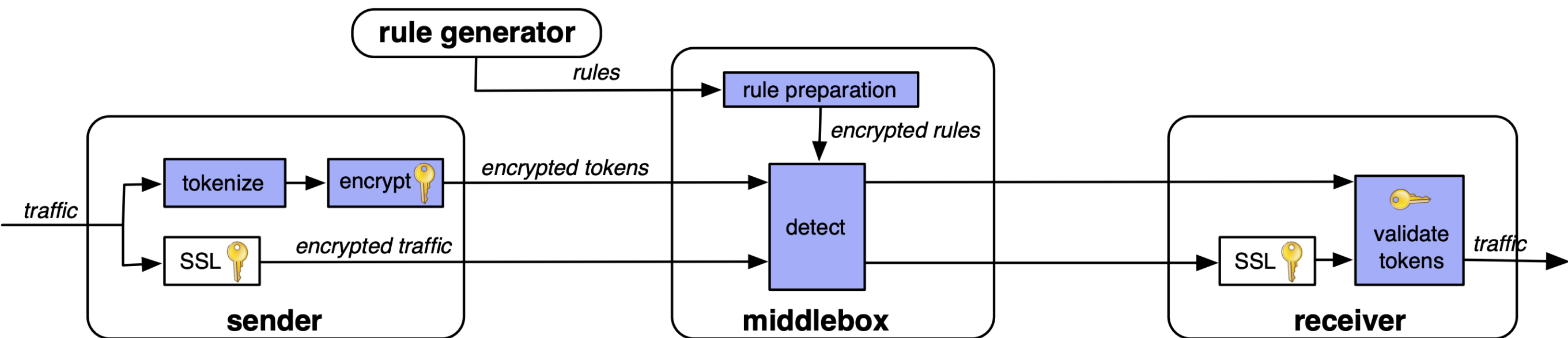
- Clients
 - Want to protect privacy from middlebox AND protection from each other
 - Requires: at least one client must be honest
- Middlebox
 - Honest but curious
 - Can only see what is necessary to enforce security policy
- Rule Generator
 - Must be trusted by both middlebox and clients
 - Cannot actually observe or alter traffic

EVALUATION HIGHLIGHTS

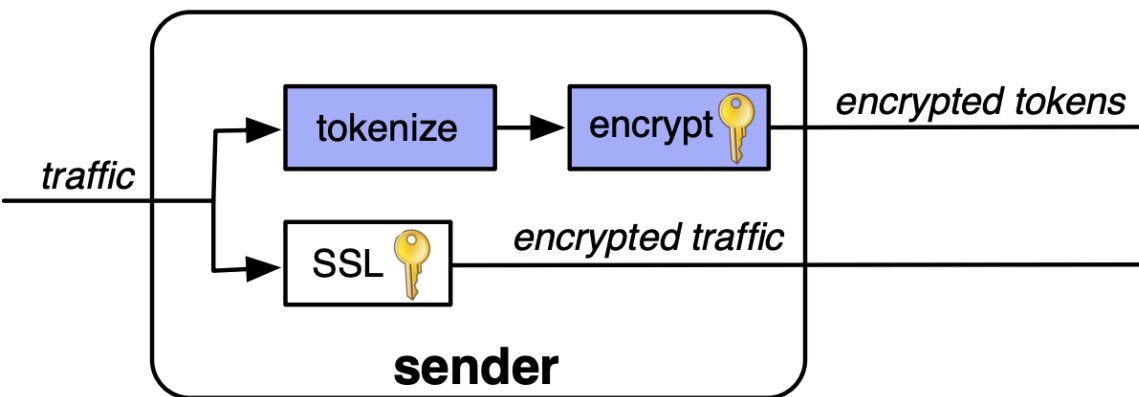
- Functionality:
 - Seems to cover the majority of use cases, esp. with protocol III
- Detection Time: Similar to existing IDS
 - 186Mbps with BlindBox (compare to Snort at 85Mbps)
- Transmission Time: Reasonable overhead
 - Page load completion time increases by 0.15-1x (ignoring setup)
- Setup Time: Very slow
 - 97 secs for 3000 rules
 - This could be OK when connections are persistent

A NAIVE IMPLEMENTATION OF BLINDBOX: PROTOCOL I

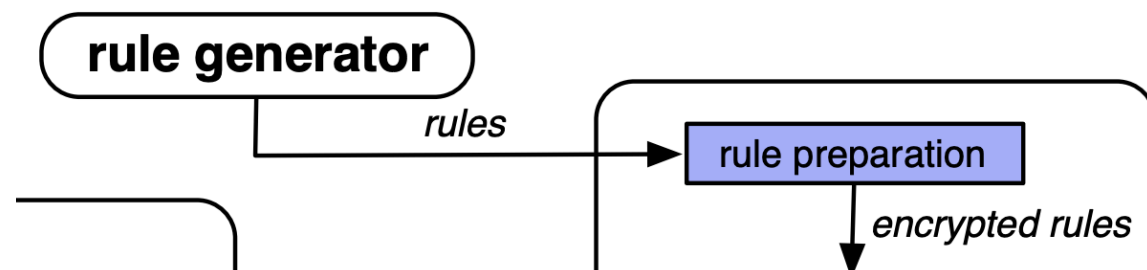
BLINDBOX: PROTOCOL I



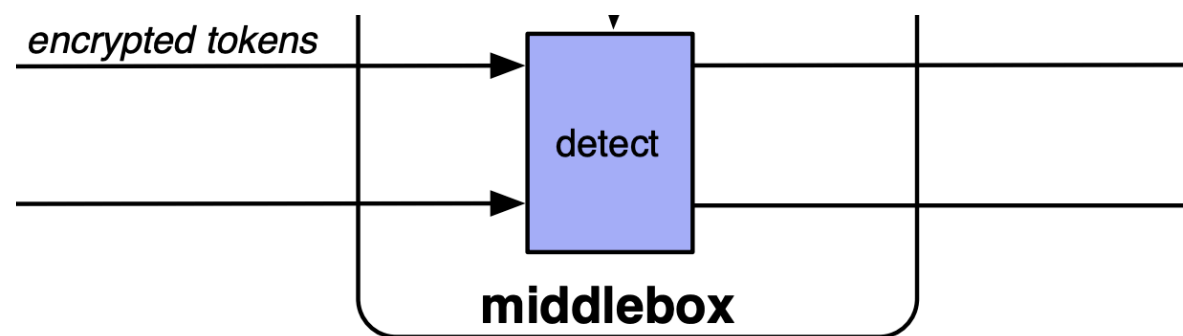
SENDER.PY



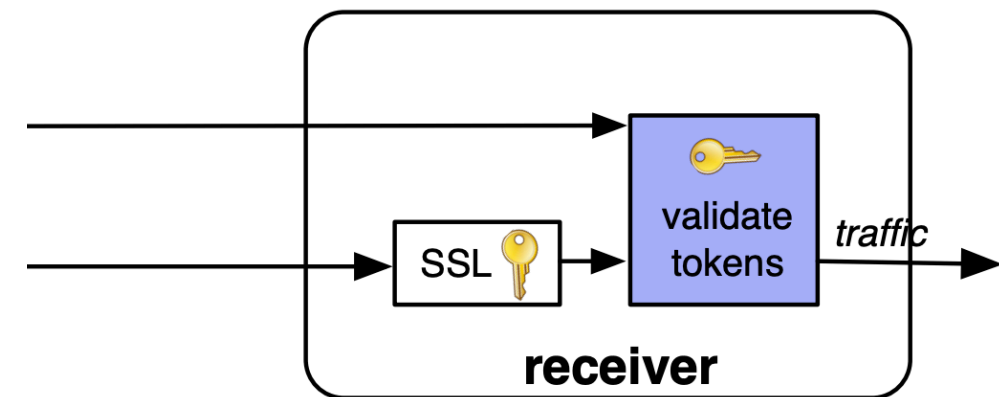
RULE_GENERATOR.PY



MIDDLEBOX.P4



RECEIVER.PY



DEMO

LIMITATIONS

- Rule detection is limited to 8 to 15 bytes only (vs. BlindBox claims to detect >8 bytes)
- Encrypted token values are hardcoded in P4
- A single token packet uses 19 bytes (vs. BlindBox uses 5 bytes per token packet)
- SSL is not implemented.
Relying on AES for the actual traffic instead.

QUESTIONS AND COMMENTS?

Thank you.