A Stateful Intrusion Detection System for World-Wide Web Servers

Giovanni Vigna
William Robertson
Vishal Kher
Richard Kemmerer

{vigna, wkr, vkher, kemm}@cs.ucsb.edu

Outline

- **Motivation and Related Work**
- System Overview
- Evaluation
- Conclusions and Future Work

Motivation

- Web servers are popular targets
  - Wide deployment
  - Exploitable custom applications
- Need to detect and mitigate impact of intrusions
  - What systems exist?

Related Work

- Network-based misuse detectors
  - Snort [Roesch]
- Application-based misuse detectors
  - Embedded intrusion detection for Apache [Almgren, Lindqvist]
  - Lightweight log analysis [Almgren et al]

Outline

- Motivation and Related Work
- **System Overview**
- Evaluation
- Conclusions and Future Work

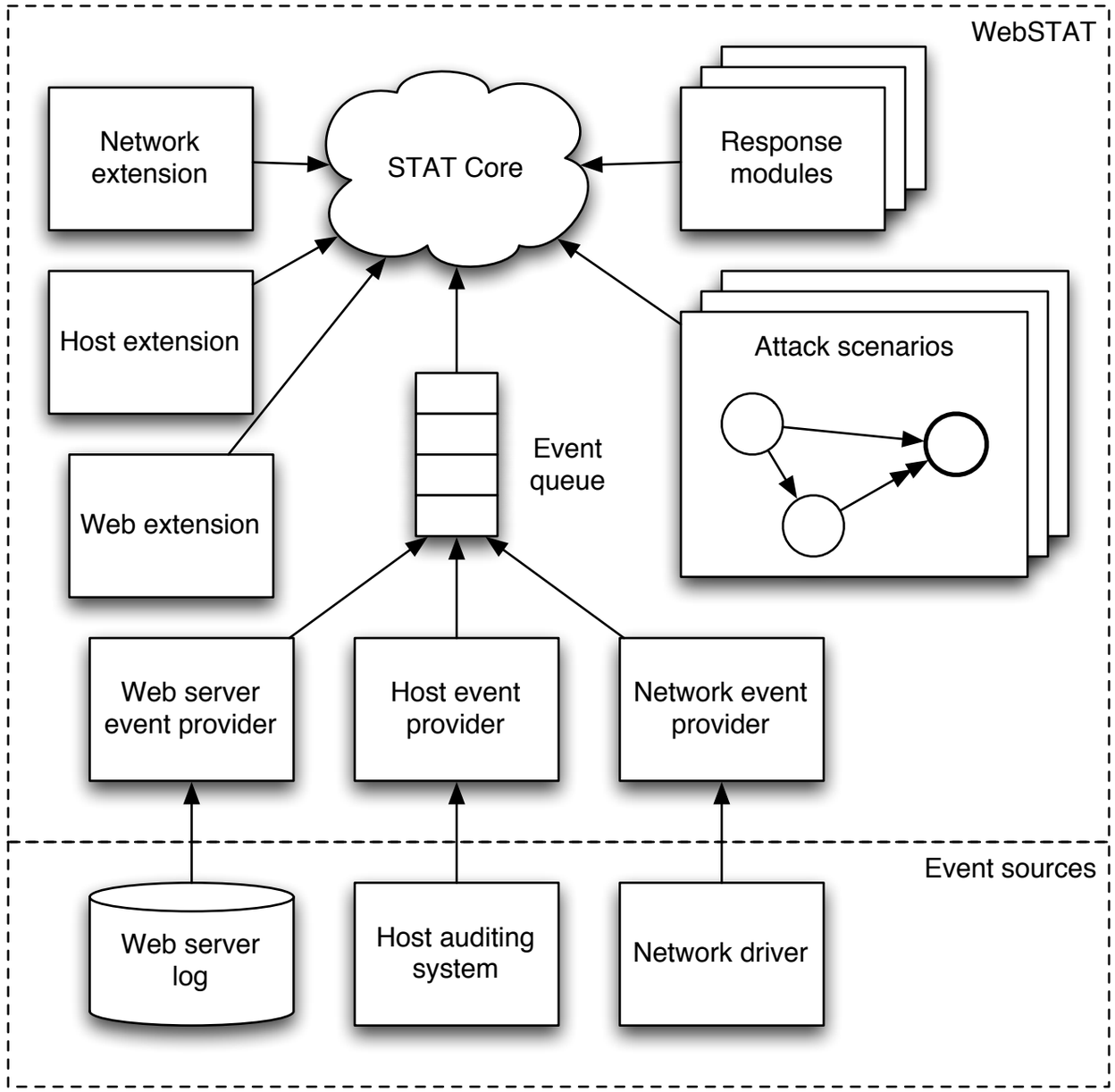
System Overview

- Based on STAT framework
- Stateful analysis on multiple event streams
- High-level modeling of complex, multi-step attacks
- Highly available, configurable
- Comprehensive set of generic signatures

STAT Framework

- Domain-independent analysis engine
- Attacks modeled as composition of *states* and *transitions*
- IDS can be assembled by composing language extensions, event providers, attack scenarios, and response functions

WebSTAT Architecture



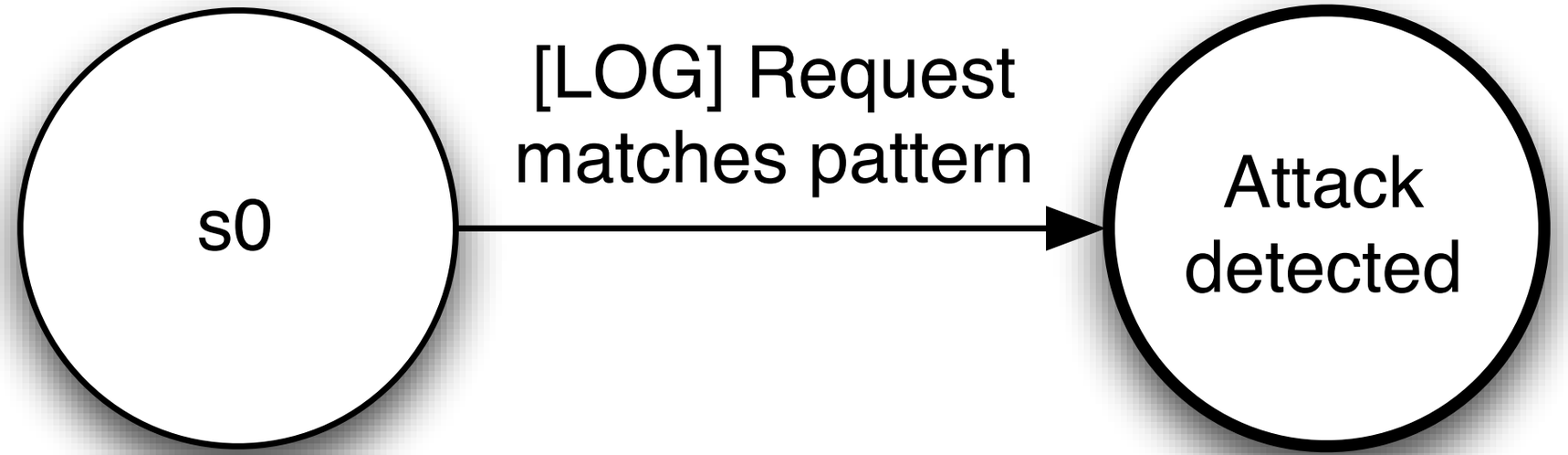
Web Extension

```
class HTTPRequest : public STATEvent

{

public:

  string request;          // Client request

  string userAgent;        // User agent

  string encodedRequest;   // Encoded request

  bool isRequestEncoded;   // Encoded flag

};
```

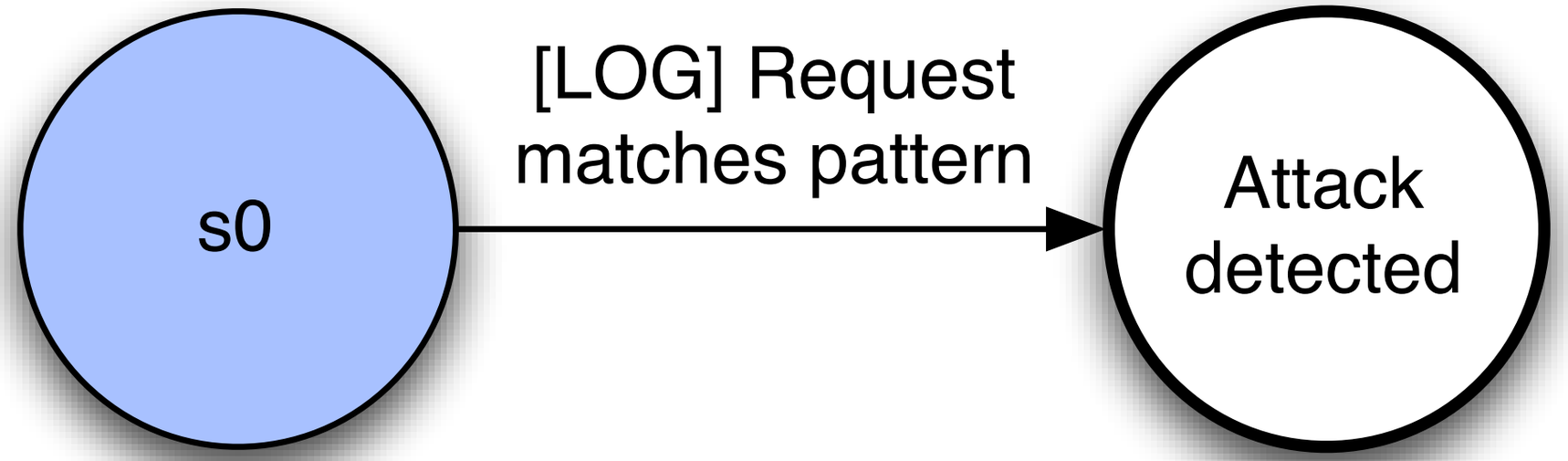
Web Event Provider

- Log-based event provider
  - Parses Common or Extended Log Format (CLF/ELF)
  - Creates and inserts HTTPRequest events into STAT core
- Network, host-based event providers

Pattern Matching



Pattern Matching

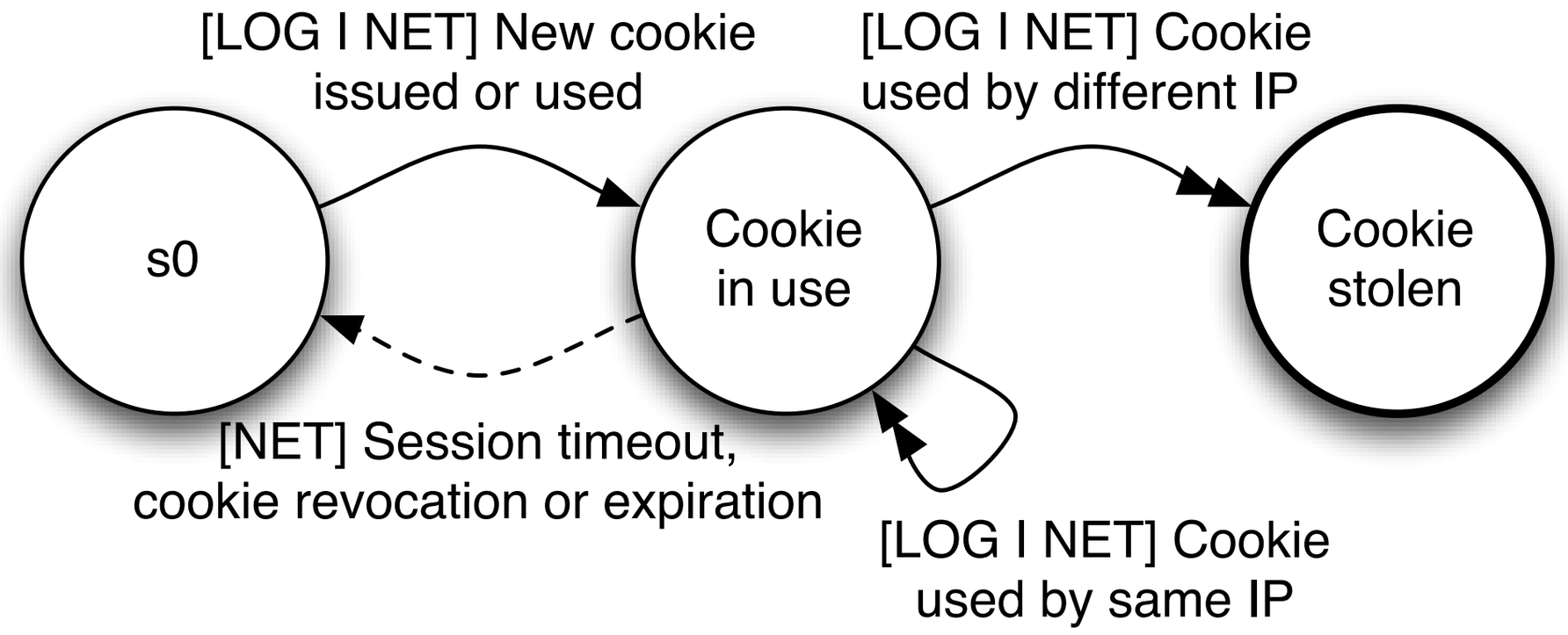


Pattern Matching

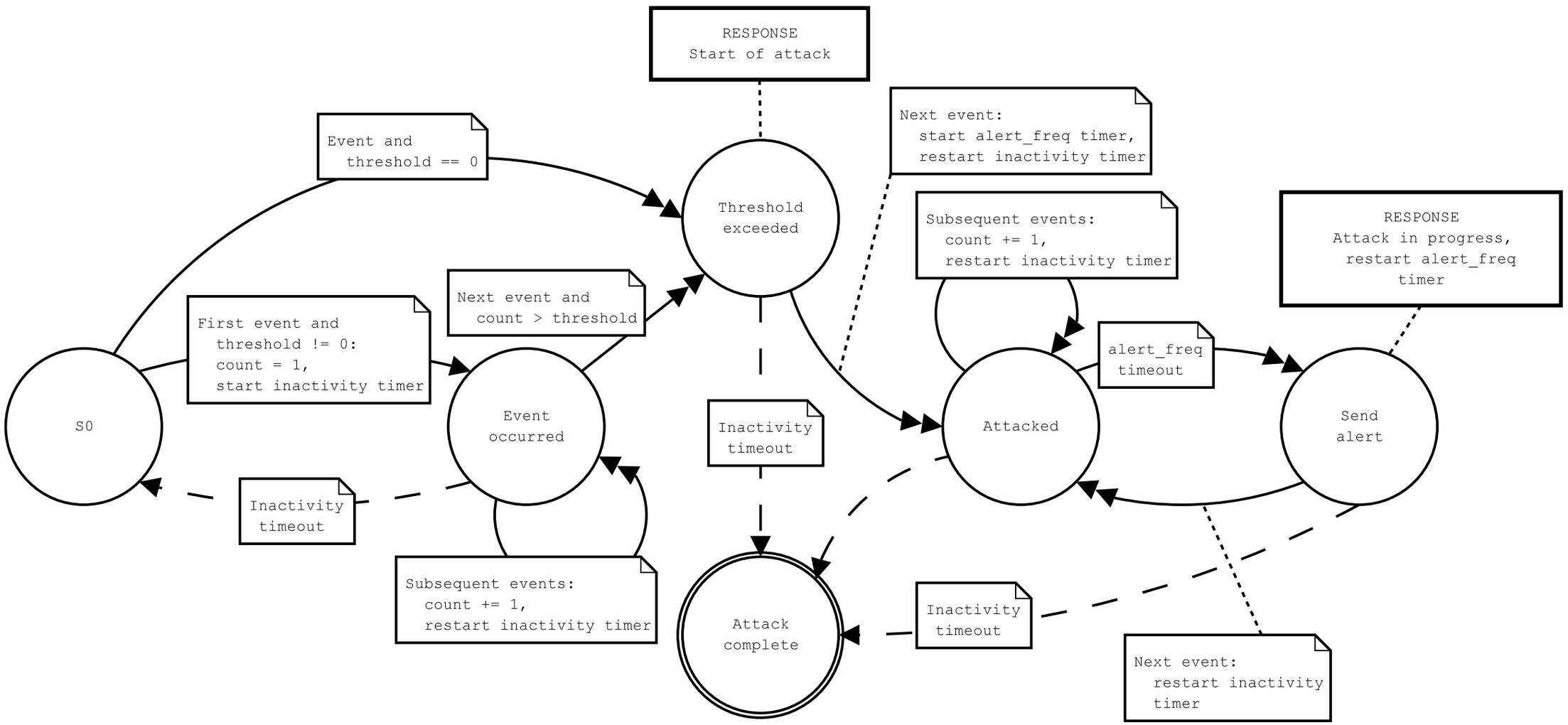
Pattern Matching

Pattern Matching

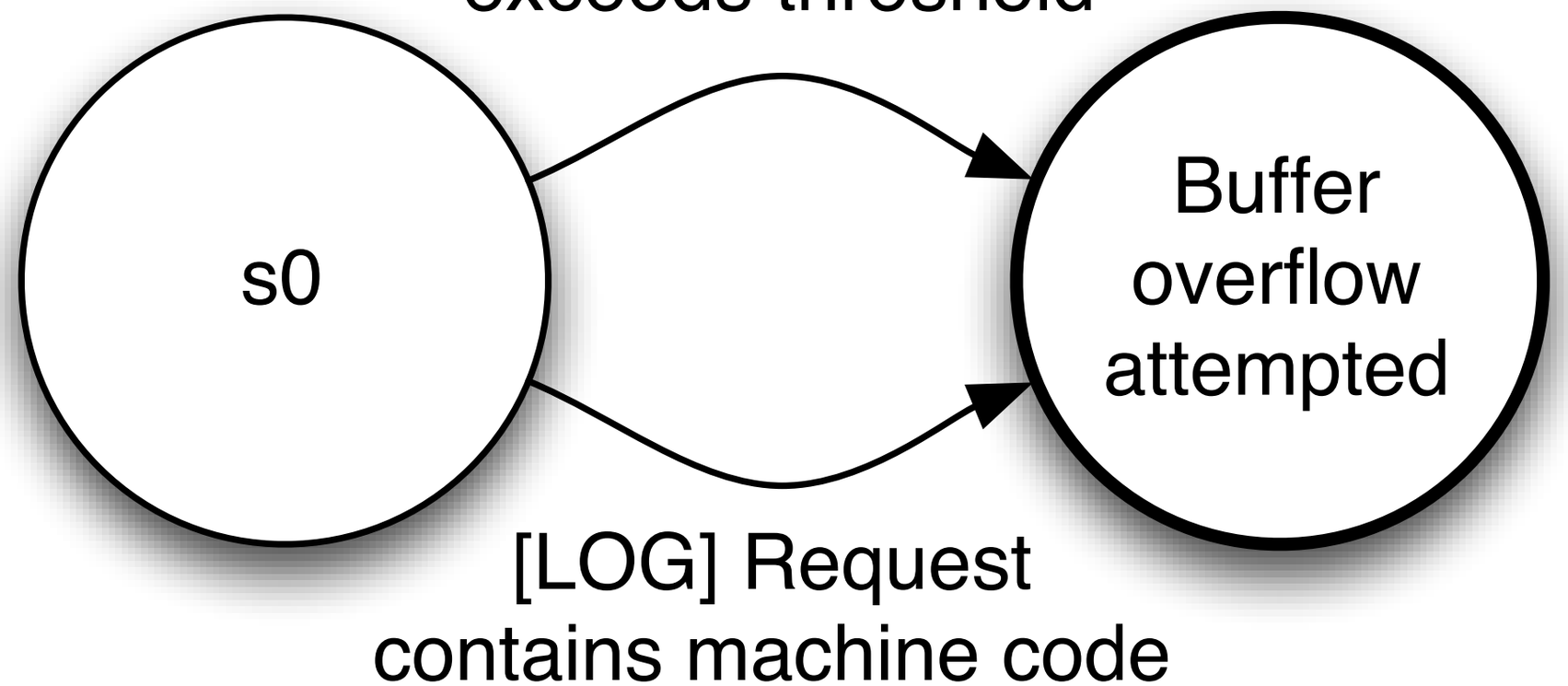
Cookie Stealing



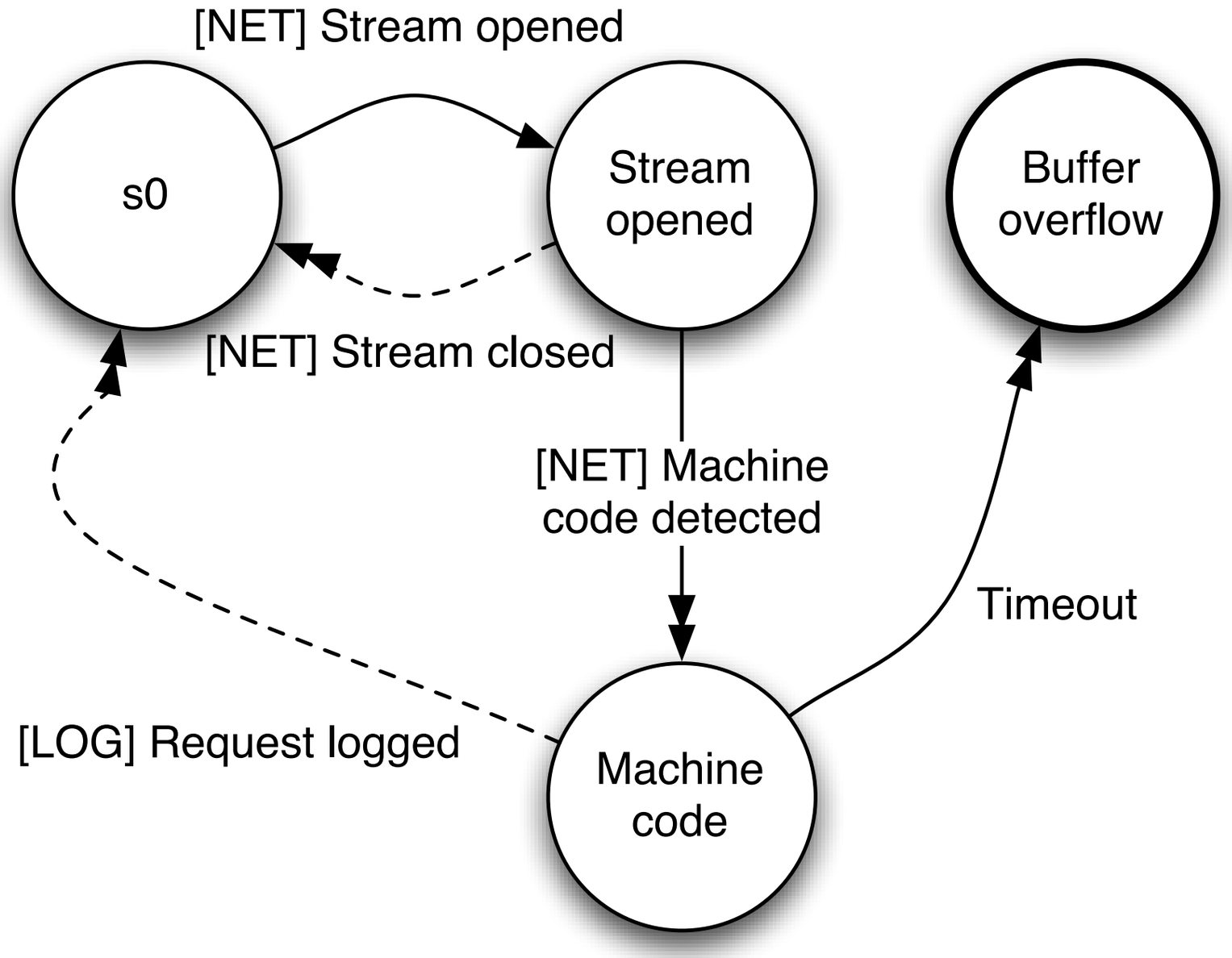
A Non-Trivial Scenario



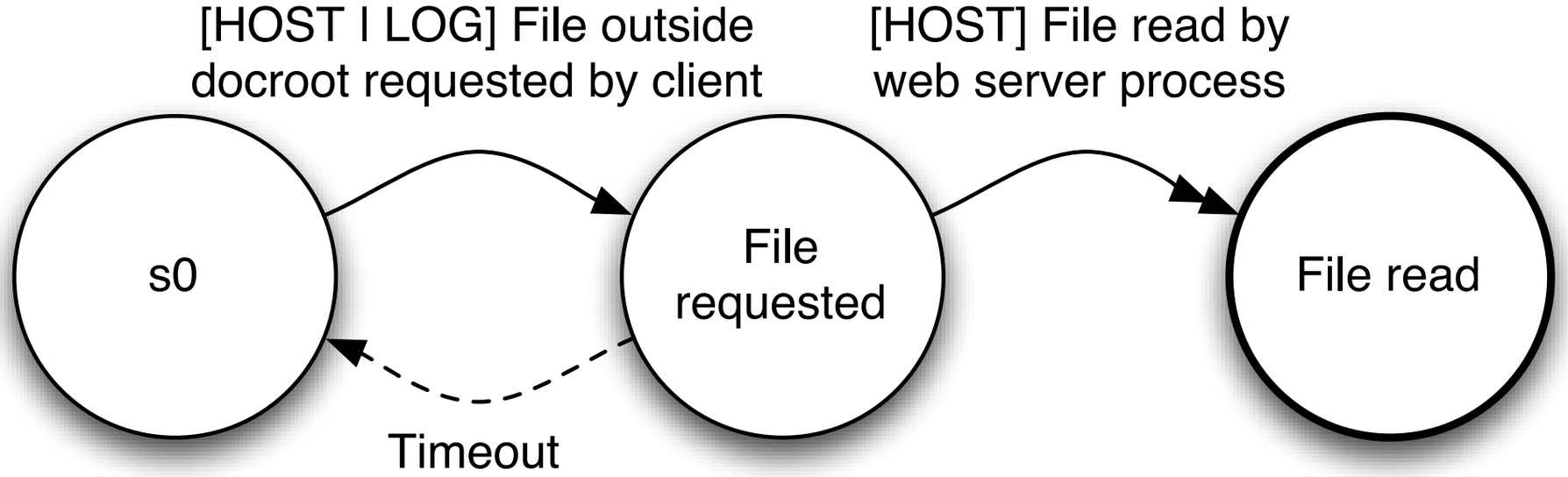
Buffer Overflow I



Buffer Overflow II



Docroot Escape



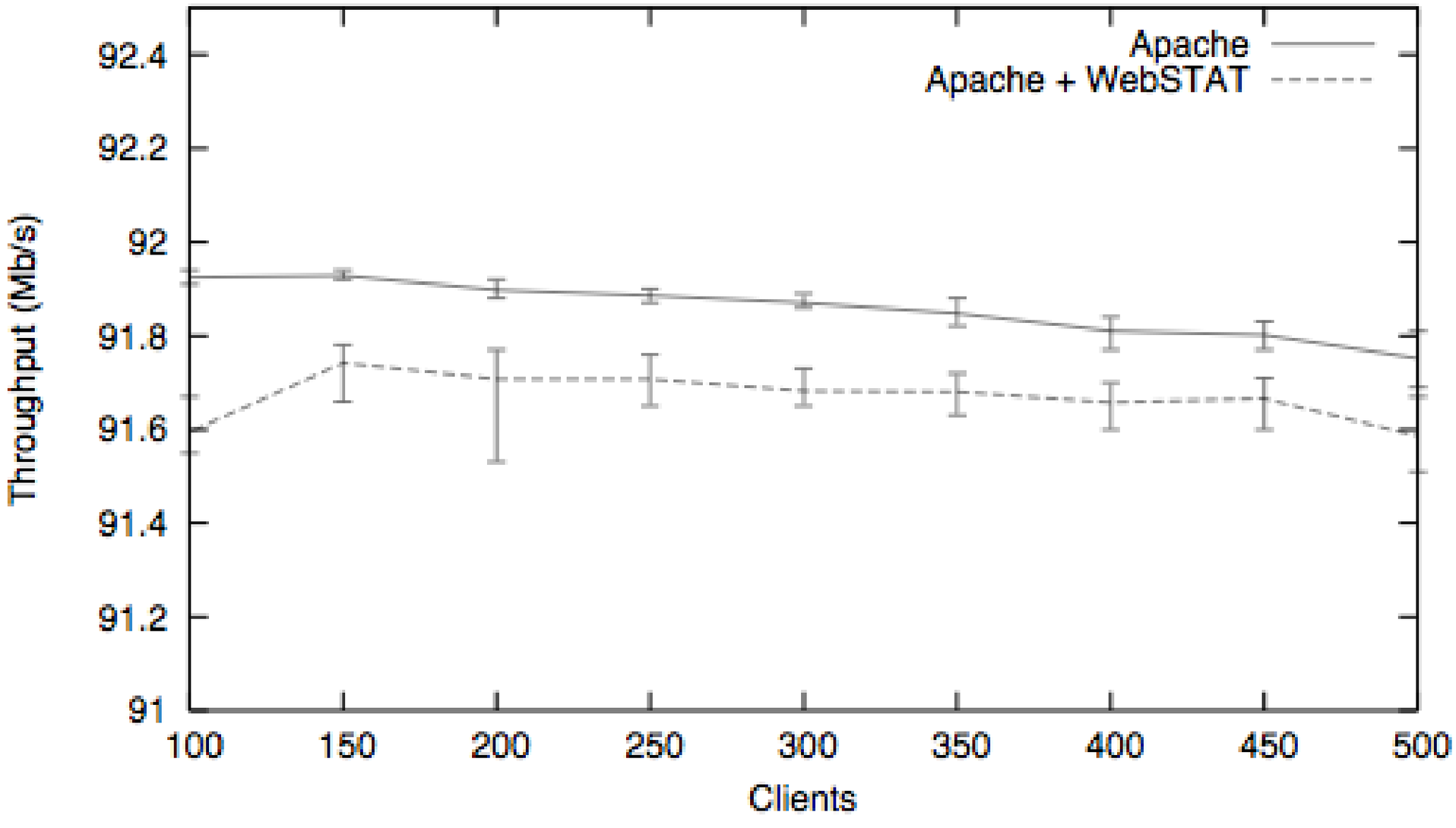
Outline

- Motivation and Related Work
- System Overview
- **Evaluation**
- Conclusions and Future Work

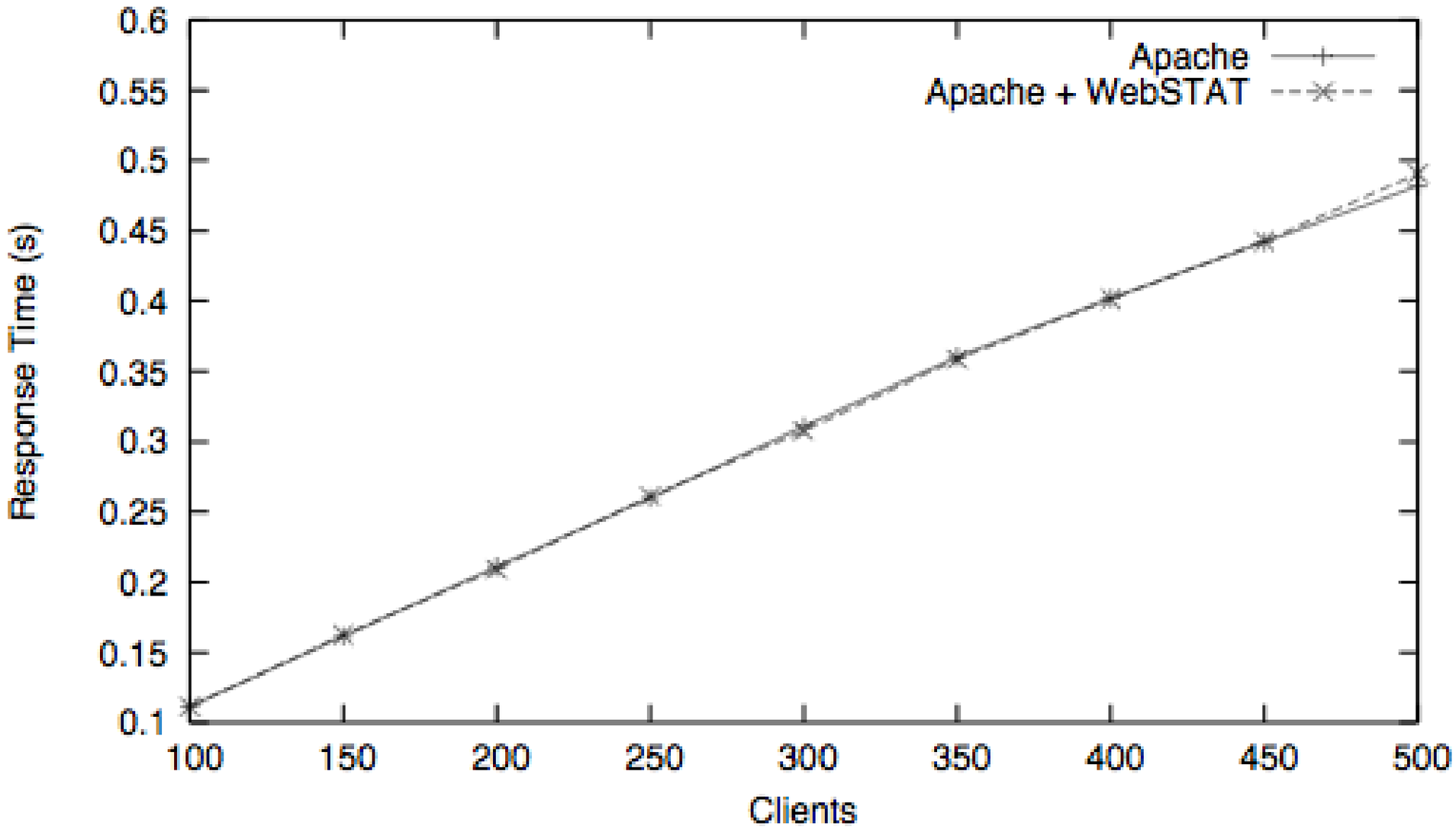
Evaluation

- Evaluate performance in production setting
- Experimental setup
  - Apache 2.0.40 / RedHat Linux 8.0
  - WebStone 2.5

Throughput


Apache
Apache + WebSTAT
Throughput (Mb/s)
92.4
92.2
92
91.8
91.6
91.4
91.2
91
100
150
200
250
300
350
400
450
500
Clients

Response Time



Outline

- Motivation and Related Work
- System Overview
- Evaluation
- **Conclusions and Future Work**

Future Work

- Develop more multi-domain attack scenarios
- Integrate anomaly-detection component to automatically generate new signatures

Conclusion

- Intrusion detection for web servers can be:
  - stateful, modeling complex attacks
  - highly configurable with no downtime
  - high performance with little overhead
  - deployed in production environments
- http://www.cs.ucsb.edu/~rsg/STAT/