CERIAS
The Center for Education and Research in Information Assurance and Security

pSigene: Webcrawling to Generalize SQLi Signatures

Gaspar Modelo-Howard, Fahad Arshad, Chris Gutierrez, Saurabh Bagchi, Yuan Qi

Problem Statement

- Misuse-based detection systems use signatures of attacks to detect malicious activity, which require to be continuously updated
- Current approach to create and update signatures is manual
- Signatures to improve detection systems, are necessary to complement prevention mechanisms

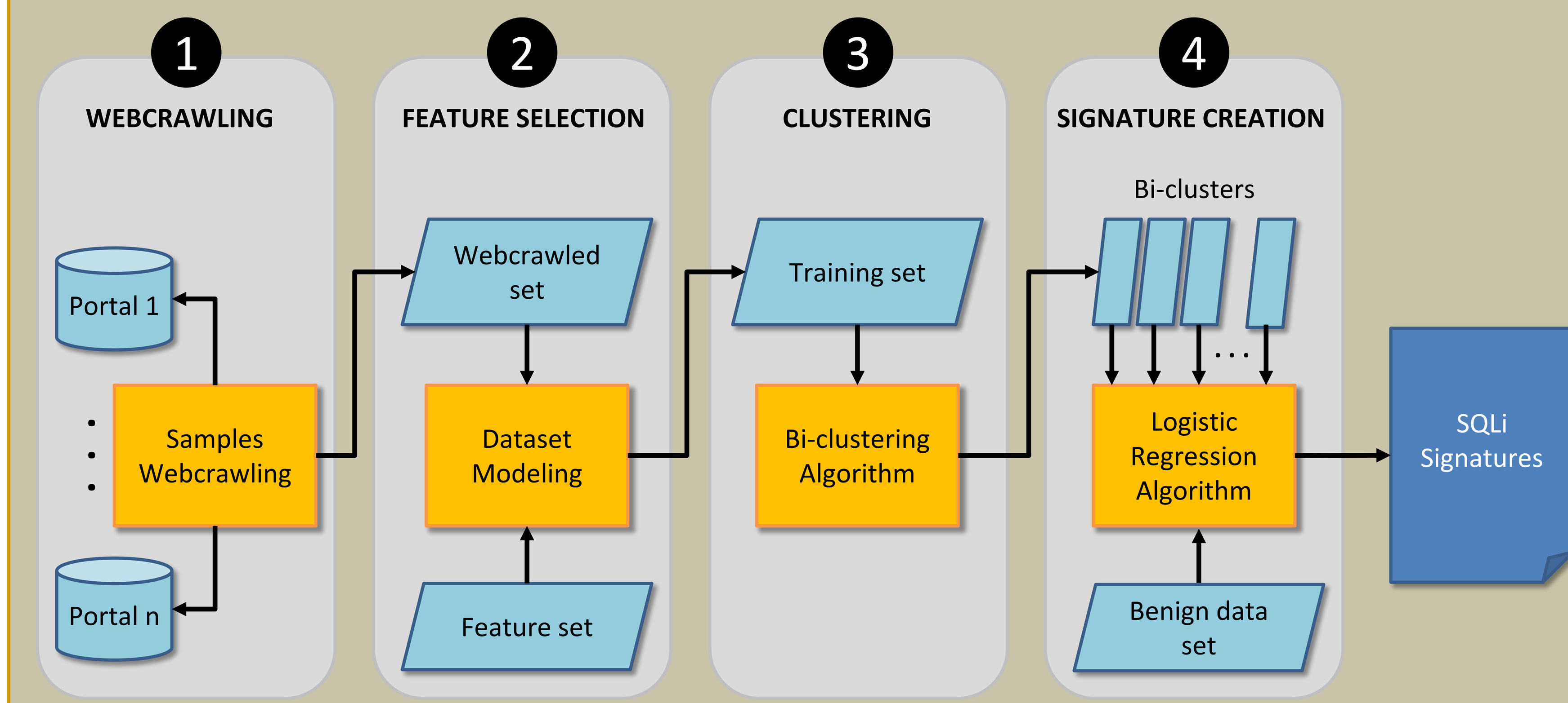
Specific Goals

- Define process to ***automatically generate detection signatures*** by performing data mining on attack samples
- Create **generalized signatures**, matching for attacks and its variations

Proposed Solution

- Framework for the automatic creation of generalized signatures represented as collection of regular expressions, by applying a sequence of two data mining techniques to a corpus of attack samples
- Solution suggests number of signatures necessary to detect attacks, while helping reduce size of signatures
- We demostrate our solution specifically with SQL injection (SQLi) attacks, which have been very dominant in the last couple of years

pSigene Architecture

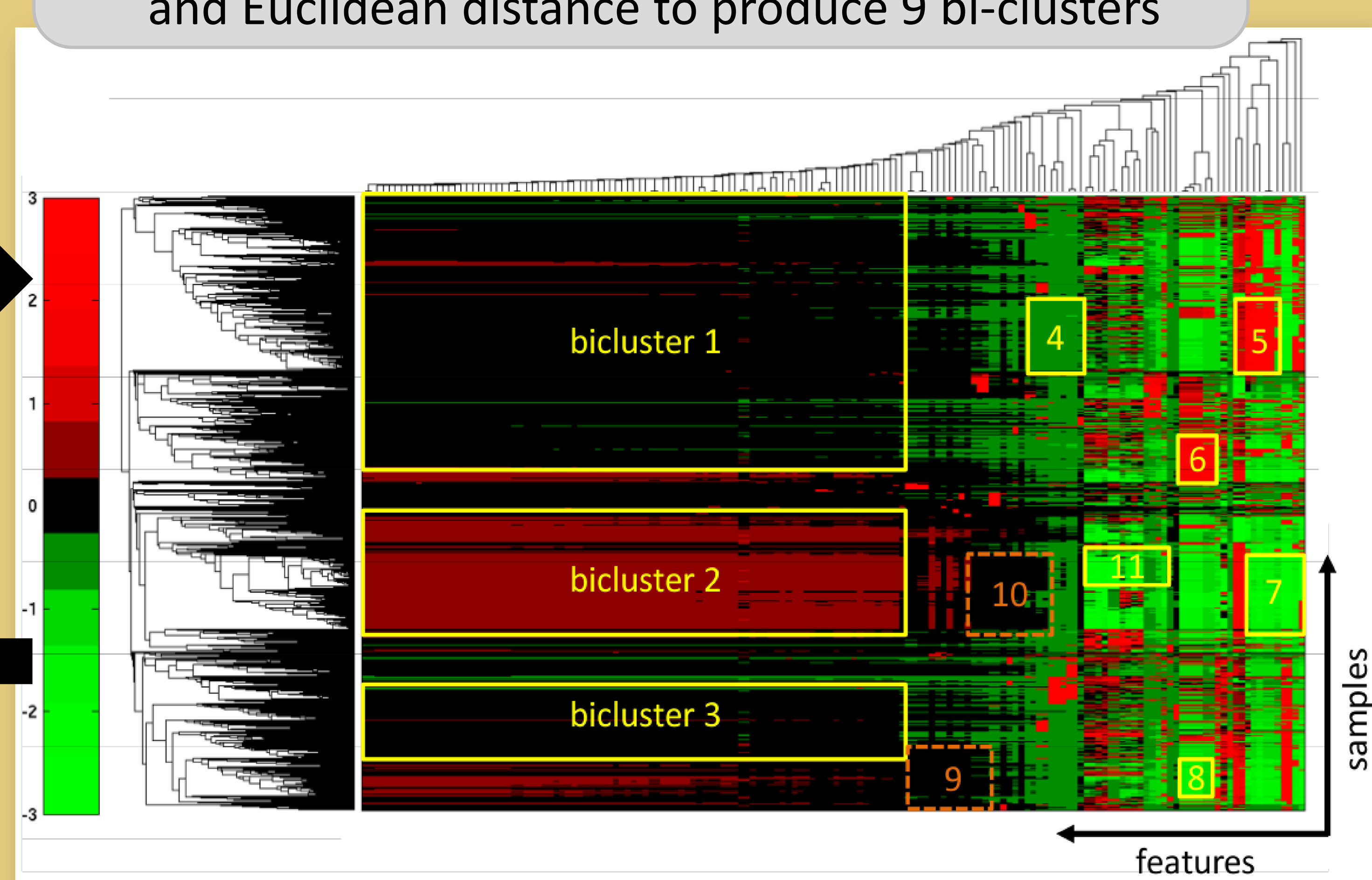


pSigene (probabilistic Signature Generation) follows a four-step process:

1. WEBCRAWLING: Search cybersecurity portals to collect attack samples
2. FEATURE SELECTION: Extract a rich set of features from attack samples and detection signatures
3. CLUSTERING: Apply bi-clustering technique to samples, identifying distinctive features for each resulting bi-cluster
4. SIGNATURE CREATION: Generate generalized signatures, one for each bi-cluster, using logistic regression modeling

Experimental Results

1. Collected over 30k SQLi attacks samples from 2 cybersecurity portals
2. Characterized each sample using set of 159 features from 3 sources: SQL reserved words, NIDS/WAF SQLi signatures, and SQLi reference documents
3. Performed a 2-way hierarchical agglomerative clustering (HAC) algorithm, using UPGMA and Euclidean distance to produce 9 bi-clusters



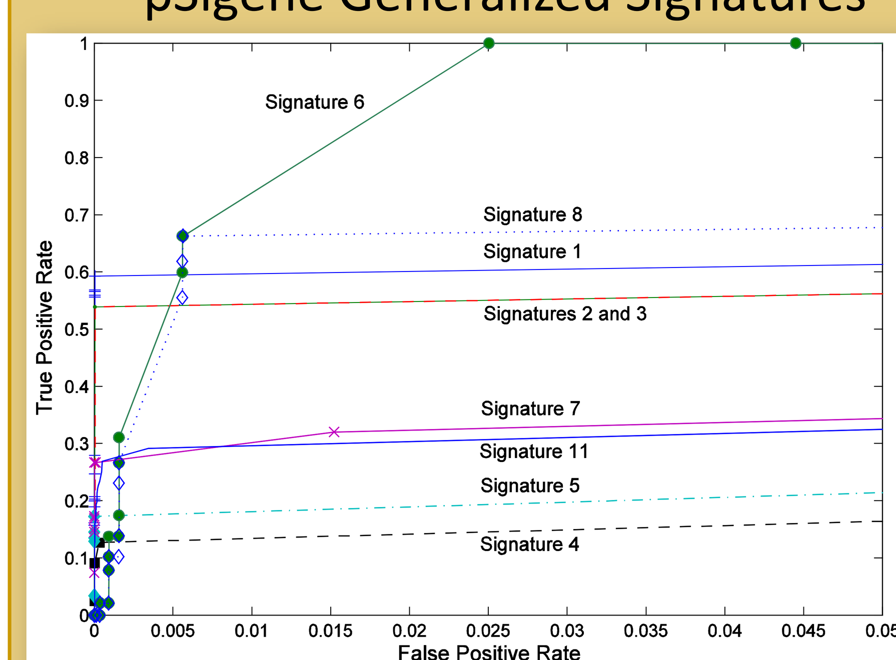
4. Generated 9 generalized signatures, one for each bi-cluster b_j, of the form:

$$Signature(b_j) = \frac{1}{1+e^{-(\Theta_j^T F_j)}} < threshold_j$$

- Each signature is a probabilistic classifier

Evaluation

- Test Set: 1.4M (benign) and 7.2k (malicious) HTTP GET requests
- ROC Curves for each of the pSigene Generalized Signatures



- Accuracy Comparison between Different SQLi Rulesets

RULES	TPR(%)	FPR(%)
Bro	73.23	0.00
Snort – Emerging Threats	79.55	0.1742
ModSecurity	96.07	0.0515
pSiGene (9 rules)	86.53	0.037
pSiGene (7 rules)	82.72	0.016

pSigene Example: Signature 6

- Signatures were implemented in Bro NIDS with function that returned number of times a feature was found in a HTTP request (count_all($f_{i,j}$, req_{HTTP}))

`“<=>|r?like|sounds+like|regex“` `“=[-0-9%]*“`

$$\Theta_6^T F_6 = -3.761 + 0.261\, f_{6,53} - 0.117\, f_{6,28} + 0.262\, f_{6,37} + 0.261\, f_{6,36} + 0.262\, f_{6,25} + 0.708\, f_{6,32}$$

`“[\?&][^\s\t\x00-\x37\|]+?“` `“([^a-zA-Z&]+)?&|exists“` `“=“` `“\)?;“`

CERIAS

PURDUE UNIVERSITY