Audit Data Reduction Using Neural Networks and Support Vector Machines

By

Srinivas Mukkamala, Andrew Sung

Presented At

The Digital Forensic Research Conference

DFRWS 2002 USA Syracuse, NY (Aug 6th - 9th)

Feature Ranking and Selection for Intrusion Detection using Support Vector Machines

Srinivas Mukkamala & Andrew H. Sung

Computer Science Department

New Mexico Tech

New Mexico Tech
SCIENCE-ENGINEERING-RESEARCH-UNIVERSITY

Intrusion Data

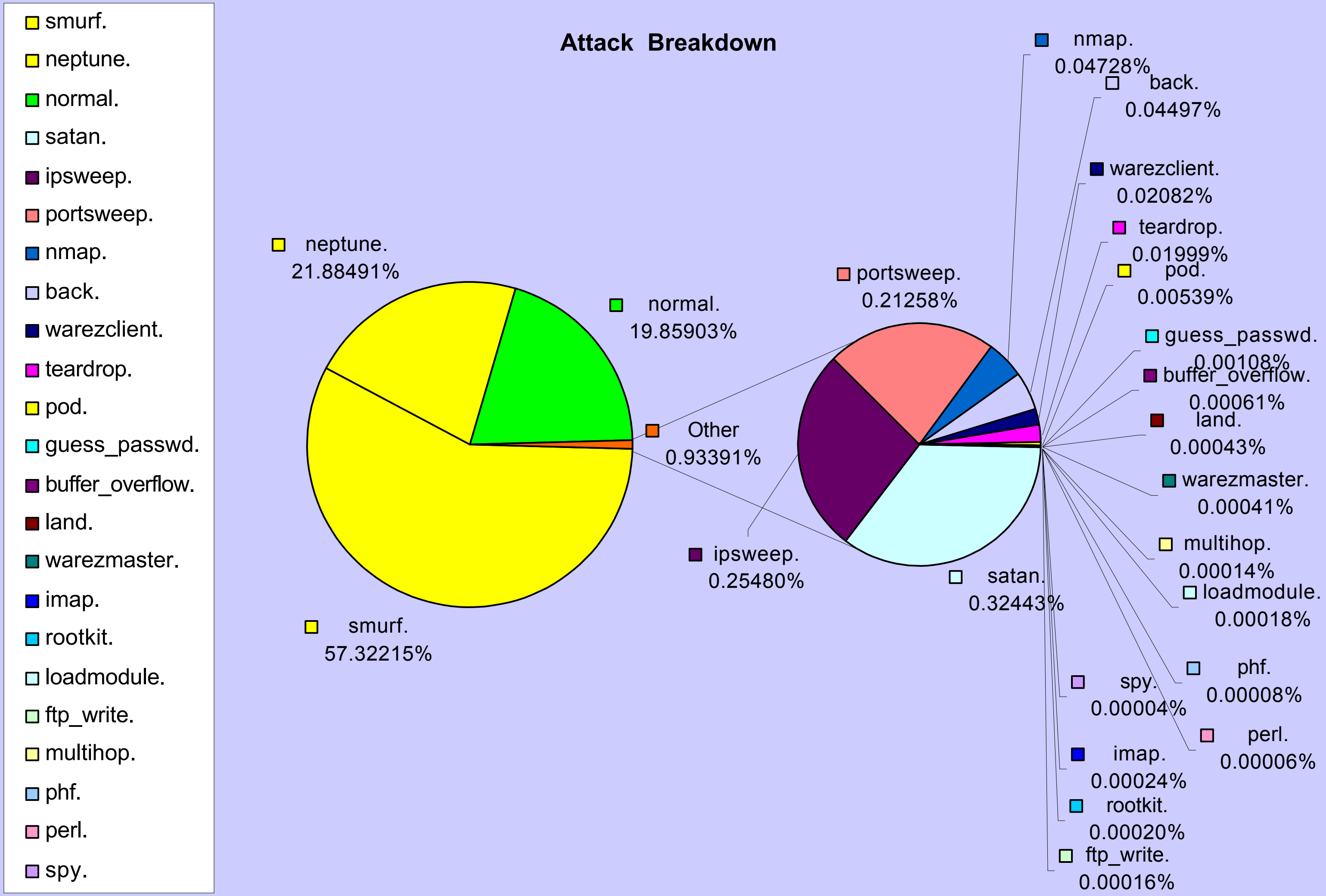
- Raw TCP/IP dump data collected form a network by simulating a typical U.S. Air Force LAN.
- For each TCP/IP connection, 41 various quantitative and qualitative features were extracted.

Attack Classes

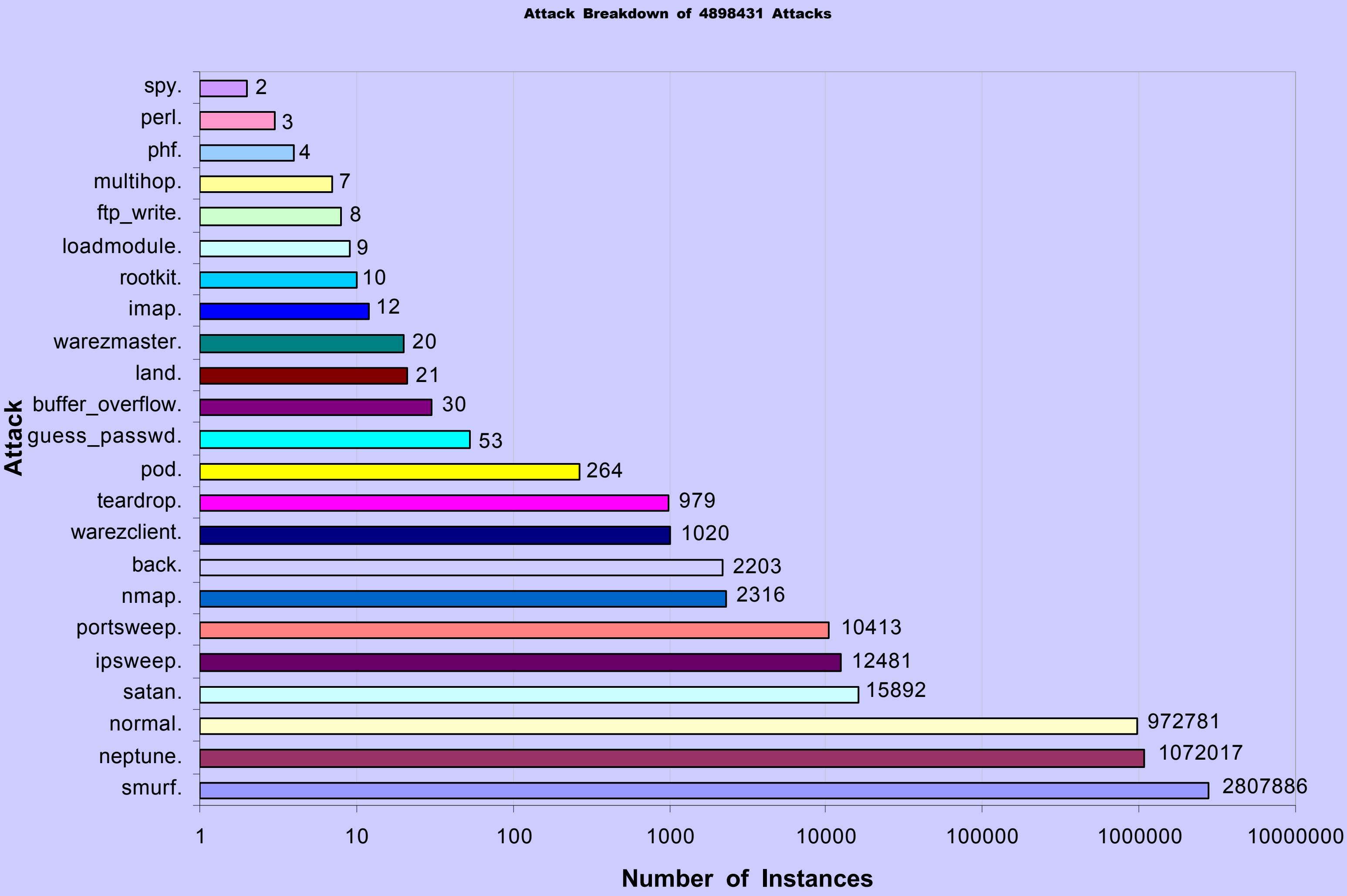
Attacks fall into four main classes:

- Probing: surveillance and other probing.
- DOS: denial of service.
- U2R: unauthorized access to local super user (root) privileges.
- R2L: unauthorized access from a remote machine.

DARPA Data


Attack Breakdown
smurf.
neptune.
normal.
satan.
ipsweep.
portsweep.
nmap.
back.
warezclient.
teardrop.
pod.
guess_passwd.
buffer_overflow.
land.
warezmaster.
imap.
rootkit.
loadmodule.
ftp_write.
multihop.
phf.
perl.
spy.
neptune. 21.88491%
normal. 19.85903%
Other 0.93391%
smurf. 57.32215%
portsweep. 0.21258%
ipsweep. 0.25480%
satan. 0.32443%
nmap. 0.04728%
back. 0.04497%
warezclient. 0.02082%
teardrop. 0.01999%
pod. 0.00539%
guess_passwd. 0.00108%
buffer_overflow. 0.00061%
land. 0.00043%
warezmaster. 0.00041%
multihop. 0.00014%
loadmodule. 0.00018%
phf. 0.00008%
perl. 0.00006%
spy. 0.00004%
imap. 0.00024%
rootkit. 0.00020%
ftp_write. 0.00016%

DARPA Data



New Mexico Tech
SCIENCE-ENGINEERING-RESEARCH-UNIVERSITY

Support Vector Machines

- Learning systems that use a hypothesis space of **linear functions** in a **high dimensional** feature space.
- Trained with a learning algorithm from optimisation theory.
- Implements a hyperplane to perform a linear (2-class) separation.

Support Vector Classification

- Consider a 2 class problem

$$F\,(\mathrm{x}\,) = \begin{array}{l} -1\text{: class A} \\ +1\text{: class B} \end{array}$$

The Feature Selection Problem

- Modeling an unknown function of a number of variables (features) based on data
- Relative significance of variables are unknown, they may be
    - Important variables
    - Secondary variables
    - Dependent variables
    - Useless variables

The Feature Selection Problem

- Which features are truly important?
- Difficult to decide due to:
  - Limited amount of data
  - Lack of algorithm
- Exhaustive analysis requires 2^n experiments (n = 41 in DARPA data).
- Need an empirical method.

Performance-Based Feature Ranking Method

- Delete one feature at a time.
- Use same training & testing sets (SVM & NN).
- If performance decreases, then feature is important.
- If performance increases, then feature is insignificant.
- If performance unchanges, then feature is secondary.

Performance-Based Feature Ranking: Procedure

- Compose the training and testing set;

for each feature *do* the following

- Delete the feature from the training and the testing data;
- Use the resultant data set to train the classifier;
- Analyze the performance of the classifier using the test set, in terms of the selected performance criteria;
- Rank the importance of the feature according to the rules;

IDS Feature Ranking: Performance Factors

- Effectiveness.
- Training time.
- Testing time.
- False Positive Rate.
- False Negative Rate.
- Other relevant measures.

Feature Ranking: Sample Rules
Support Vector Machines

A (accuracy), LT (learning time), TT (testing time).

- If A ⌈ and LT ⌈ and TT ⌈ , then feature is insignificant.
- If A ⌈ and LT ⌈ and TT ⌈ , then feature is important.
- If A ⌈ and LT ⌈ and TT ⌈ , then feature is important.

  .

  .

  .

- Otherwise, feature is secondary.

Feature Ranking: Sample Rules
Neural Networks

A (accuracy), FP (false positive rate), FN (false negative rate).

- If A ⌈ and FP ⌈ and FN ⌈ , then feature is insignificant.
- If A ⌈ and FP ⌈ and FN ⌈ , then feature is important.
- If A ⌈ and FP ⌈ and FN ⌈ , then feature is important.

•

•

•

- Otherwise, feature is secondary.

Rule Set

1. ***If*** *accuracy* decreases ***and*** training time increases ***and*** testing time decreases, ***then*** the feature is important
2. ***If*** *accuracy* decreases ***and*** training time increases ***and*** testing time increases, ***then*** the feature is important
3. ***If*** *accuracy* decreases ***and*** training time decreases ***and*** testing time increases, ***then*** the feature is important
4. ***If*** *accuracy* unchanges ***and*** training time increases ***and*** testing time increases, ***then*** the feature is important
5. ***If*** *accuracy* unchanges ***and*** training time decreases *and* testing time increases, ***then*** the feature is secondary

Rule Set

6. ***If*** *accuracy* unchanges ***and*** training time increases ***and*** testing time decreases, ***then*** the feature is secondary
7. ***If*** *accuracy* unchanges ***and*** training time decreases ***and*** testing time decreases, ***then*** the feature is unimportant
8. ***If*** *accuracy* increases ***and*** training time increases ***and*** testing time decreases, ***then*** the feature is secondary
9. ***If*** *accuracy* increases ***and*** training time decreases ***and*** testing time increases, ***then*** the feature is secondary
10. ***If*** *accuracy* increases ***and*** training time decreases ***and*** testing time decreases, ***then*** the feature is unimportant

Performance-Based Feature Ranking Advantages

- General applicability (ANNs, SVMs, etc.)
- Linear complexity (requiring only O(n) experiments).
- Tuning of rules to improve results.
- Multi-level ranking is possible.

Performance-Based Feature Ranking Results

Important Secondary Unimportant

Normal	1,3,5,6,8-10,14,15,17,20-23,25- 29,33,35,36,38, 39 41, 2,4,7,11,12,16,18,19, 24,30,31,34,37,40, 13,32
Probe	3,5,6,23,24,32,33, 1,4,7-9,12-19,21,22,25-28, 34-41, 2,10,11,20,29,30,31,36,37
DOS	1,3,5,6,8,19,23-28,32,33,35,36,38-41, 2,7,9-11, 14, 17,20,22,29,30,34,37, 4,12,13,15,16,18,19,21,31
U2R	5,6,15,16,18,25,32,33, 7,8,11,13,17,19-24,26,30, 36-39, 9,10,12,14,27,29,31,34,35,40,41
R2L	3,5,6,24,32,33, 2,4,7-23,26-31,34-41, 1,20,25,38

SVM: Using All 41 Features

Class	Training time (sec)	Testing time (sec)	Accuracy	Class size 5092 : 6890
Normal	7.66	1.26	99.55%	1000:1400
Probe	49.13	2.10	99.70%	500:700
DOS	22.87	1.92	99.25%	3002:4207
U2R	3.38	1.05	99.87%	27:20
R2L	11.54	1.02	99.78%	563:563

SVM: Using Important Features

Class	No of Features	Training time (sec)	Testing time (sec)	Accuracy	Class size 5092:6890
Normal	25	9.36	1.07	**99.59%**	1000:1400
Probe	7	37.71	1.87	99.38%	500:700
DOS	19	22.79	1.84	99.22%	3002:4207
U2R	8	2.56	0.85	99.87%	27:20
R2L	6	8.76	0.73	99.78%	563:563

SVM: Using Union of Important Features of All Classes, 30 Total

Class	Training time	Testing time	Accuracy	Class size 5092:6890
Normal	7.67	1.02	99.51%	1000:1400
Probe	44.38	2.07	99.67%	500:700
DOS	18.64	1.41	99.22%	3002:4207
U2R	3.23	0.98	99.87%	27:20
R2L	9.81	1.01	99.78%	563:563

SVM: Using Important Features + Secondary Features

Class	No of Features	**Training time (sec)**	**Testing time (sec)**	**Accuracy**	**Class size 5092:6890**
Normal	39	8.15	1.22	**99.59%**	**1000:1400**
Probe	32	47.56	2.09	99.65%	**500:700**
DOS	32	19.72	2.11	99.25%	**3002:4207**
U2R	25	2.72	0.92	99.87%	**27:20**
R2L	37	8.25	1.25	**99.80%**	**563:563**

Performance Statistics
(using performance-based ranking)

🟩 All features

🟨 Important features + Secondary features

🟥 Important features

🟦 Union of important features

Performance Statistics
(using performance-based ranking)

Normal	🟨 99.59%	🟥 99.59	🟩 99.55	🟦 99.51
Probe	🟩 99.70	🟦 99.67	🟨 99.65	🟥 99.38
DOS	🟩 99.25	🟨 99.25	🟥 99.22	🟦 99.22
U2R	🟩 99.87	🟥 99.87	🟨 99.87	🟦 99.87
R2L	🟨 99.80	🟩 99.78	🟥 99.78	🟦 99.78

Feature Ranking using Support Vector Decision Function

$$\mathbf{F(X) = \lceil \quad W_i X_i + b}$$

- F(X) depends on the contribution of $\mathbf{W_i X_i}$
- Absolute value of $\mathbf{W_i}$ measures the strength of classification of classification

Feature Ranking using Support Vector Decision Function (SVDF)

- ***if*** W$_i$ is a large positive value ***then*** the **i**th feature is a key factor for the positive class
- ***if*** W$_i$ is a large negative value ***then*** the **i**th feature is a key factor for the negative class
- ***if*** W$_i$ is a value close to zero on either the positive or negative side ***then*** the ith feature does not contribute significantly to the classification

SVM Based Feature Ranking Method

- Calculate the weights from the support vector decision function.
- Rank the importance of the features by the absolute values of the weights.
- Delete the insignificant features from the training and the testing data.
- Use the resultant data set to train the classifier.
- Analyze the performance of the classifier using the test set, in terms of the selected performance criteria (threshold values of the weights for ranking the features).

SVM Based Feature Ranking: Advantages

- Uses SVMs decision function.
- Linear complexity (requiring only O(n) experiments).
- Tuning of the ranking process by adjusting the threshold values.
- Multi-level ranking is possible.

SVM-Based Feature Ranking Results

Important Secondary

Normal	2,3,4,6,10,12,23,29,32,33,34,36, 1,5,7-9,11,13-22, 24-28,30,31,35,37-41
Probe	2,4,5,23,24,33, 1,3,6-22,25-32,34-41
DOS	23,24,25,26,36,38,39, 1-22,27-35,40,41
U2R	1,2,4,5,12,29,34, 3,6-11,13-28,30-33,35-41
R2L	1,3,32, 2,4-31,33-41

SVM: Using Important Features as ranked by SVDF

Class	No of Features	Training time (sec)	Testing time (sec)	Accuracy	Class size 5092:6890
Normal	15	3.73	.98	**99.56%**	1000:1400
Probe	12	41.44	1.63	99.35%	500:700
DOS	16	20.43	1.62	99.14%	3002:4207
U2Su	13	1.82	0.97	99.87%	27:20
R2L	6	3.24	.98	**99.72%**	563:563

SVM: Union of Important Features of All Classes : 19 Total

training : testing 5092 : 6890

Class	Training time	Testing time	Accuracy	Class size 5092:6890
Normal	4.35	1.03	99.55%	1000:1400
Probe	26.52	1.73	99.42%	500:700
DOS	8.64	1.61	99.19%	3002:4207
U2R	2.04	0.18	99.85%	27:20
R2L	5.67	1.12	99.78%	563:563

Performance Statistics (using SVM-based ranking)

🟩 All features

🟥 Important features

🟦 Union of important features

Performance Statistics
(using SVM-based ranking)

Normal	99.56%	99.55	99.55
Probe	99.70	99.42	99.35
DOS	99.25	99.19	99.14
U2R	99.87	99.87	99.85
R2L	99.78	99.78	99.78

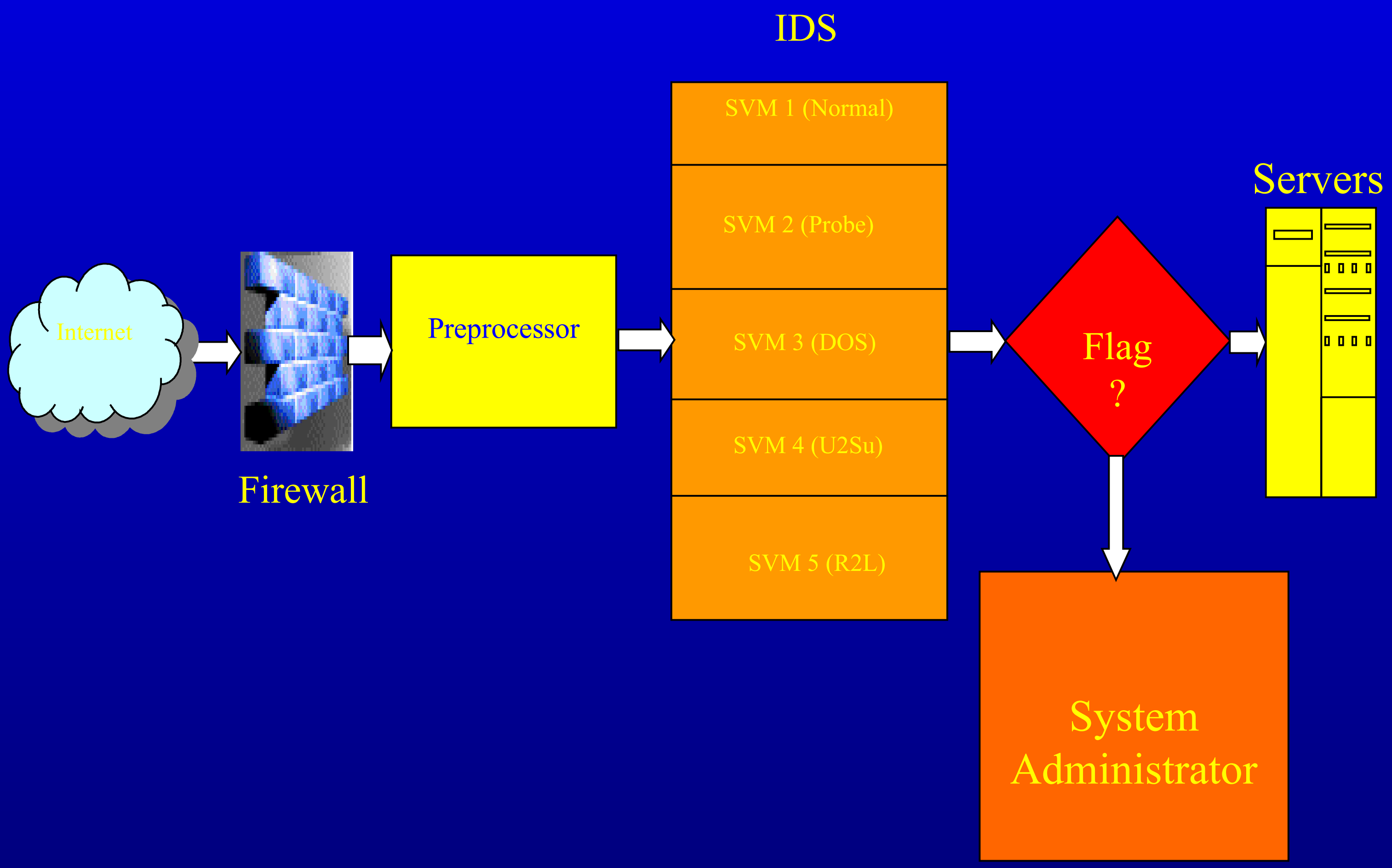
IDS Feature Ranking: Performance Factors

- Effectiveness.
- Training time.
- Testing time.
- False Positive Rate.
- False Negative Rate.
- Other relevant measures.

Two Feature Ranking Methods: Performance Summary

- Important features selected by two methods heavily overlap.
- Different levels of SVM IDS performance are achieved by
  - using all features
  - using important features
  - using union of important features
- However, the performance difference is small

A New IDS Architecture Using SVMs



Conclusions

- IDS based on SVMs.
- SVMs generally outperform NNs (cf. reference 2)
- Two methods for feature ranking of 41 inputs, for each of the 5 classes.
- Using important features give comparable performance.
- New IDS comprising 5 SVMs delivers high accuracy and faster (than NN) running time.

References

- **S. Mukkamala, G. Janowski, A. H. Sung,** *Intrusion Detection Using Support Vector Machines*, Proceedings of the High Performance Computing Symposium – HPC 2002, April 2002, pp.178-183.
- **S. Mukkamala, G. Janowski, A. H. Sung,** *Intrusion Detection Using Neural Networks and Support Vector Machines*, Proceedings of IEEE IJCNN, May 2002, pp.1702-1707.
- **Srinivas Mukkamala, Andrew Sung,** *Feature Ranking and Selection for Intrusion Detection*, Proceedings of the International Conference on Information and Knowledge Engineering – IKE 2002, June 2002, pp.503-509.