Web Server Anomaly Detection using Principle Component Analysis

NECTEC-ACE2010

Nattaya Sukityarn, KMUTT
Poj Tangamchit, KMUTT
Chavee Issariyapat, NECTEC
Panita Pongpaibool, NECTEC

27/9/10

Agenda

- Introduction

  * Motivation *Features

  * Access log and Error log *PCA

- Methodology

Click to edit Master subtitle style

- Experiment I: Determining Useful Dimensions
- Experiment II: Anomaly Detection
- Future Works

27/9/10

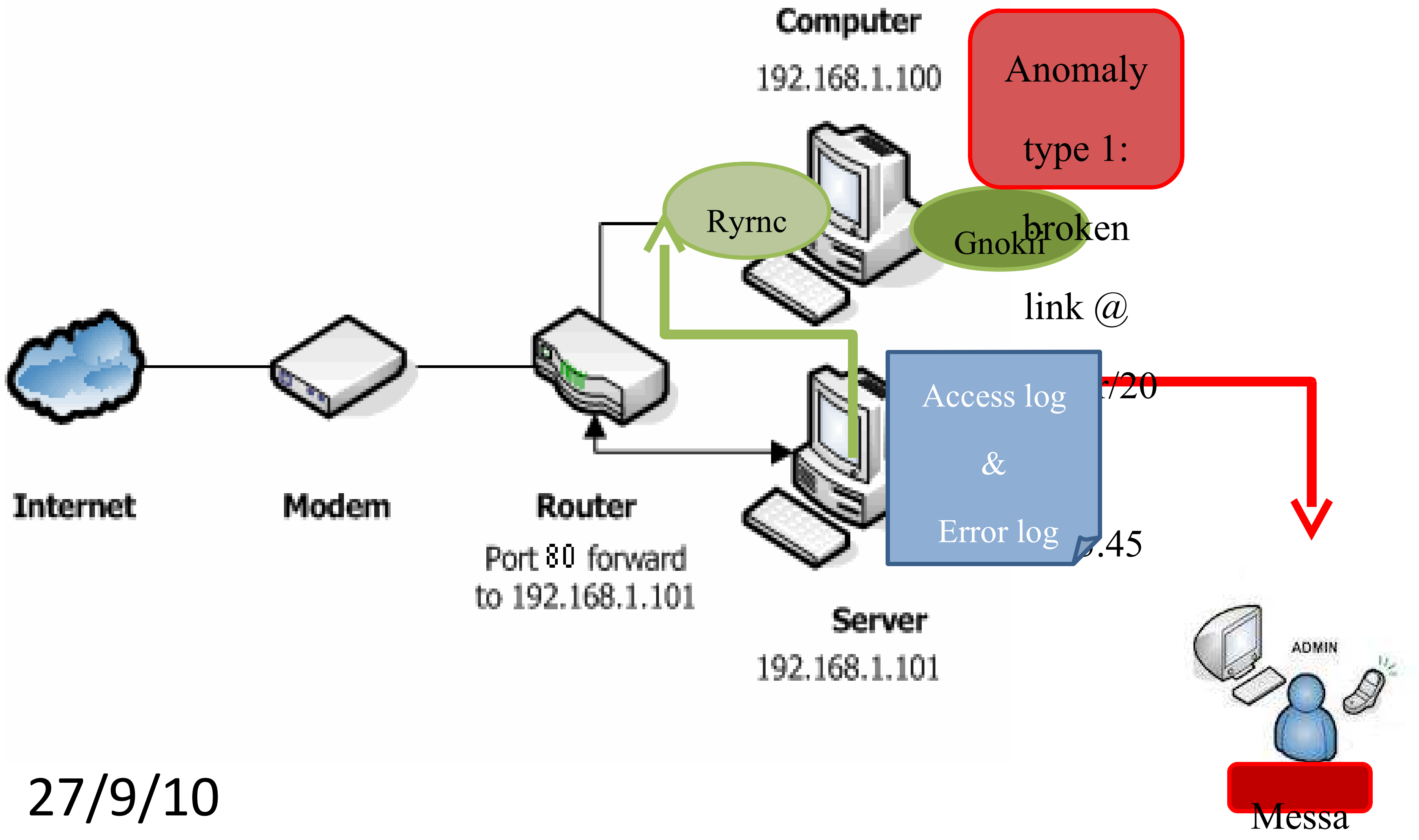
Introduction: Motivation

- Network Monitoring Systems usually used host-specific thresholds to justify host status e.g. threshold on PING round trip time
- NTL teams are developing a NMS called NetHAM.
- One of the goal is to let NetHAM analyzes network hosts and their services without using host-specific threshold.
- We start at one of the most common internet services i.e. HTTP.
- Here comes this experiment…

Click to edit Master subtitle style

27/9/10

Introduction: Motivation



Anomaly type 1: broken link @ :/20 .45

Access log & Error log

Messa

27/9/10

Introduction: Access log & Error log

Access Log : The server access log records all requests processed by the server.

```
202.44.135.34 - - [09/Sep/2003:20:30:11+0700] “Get /PROJECT/member.html HTTP/1.0” 200 19037
      1       2 3             4                  5          6              7        8     9
```

Click to edit Master subtitle style

Error Log : The server error log records any errors that it encounters in processing requests.

```
[Fri Jun 20 22:19:44 2003] [error] [client 127.0.0.1] Invalid method in request\x80L\x01\x03
          1                   2           3                         4
```

27/9/10

Introduction: Features

The ten features are as follows.

1. Number of unique clients (F1)
2. Number of request (F2)
3. Number of error events (Er)
4. Number of HTTP Unauthorized Access events (401)
5. Number of HTTP Not Found events (404)
6. Number of HTTP Forbidden events (403)
7. Number of HTTP Internal Error events (500)
8. Maximum number of requests per client (Max)
9. Minimum number of requests per client (Min)
10. Average number of request per client (Avg)

Click to edit Master subtitle style

Please note! All ten features are calculated over a 5-minute interval

Introduction: Basic Idea

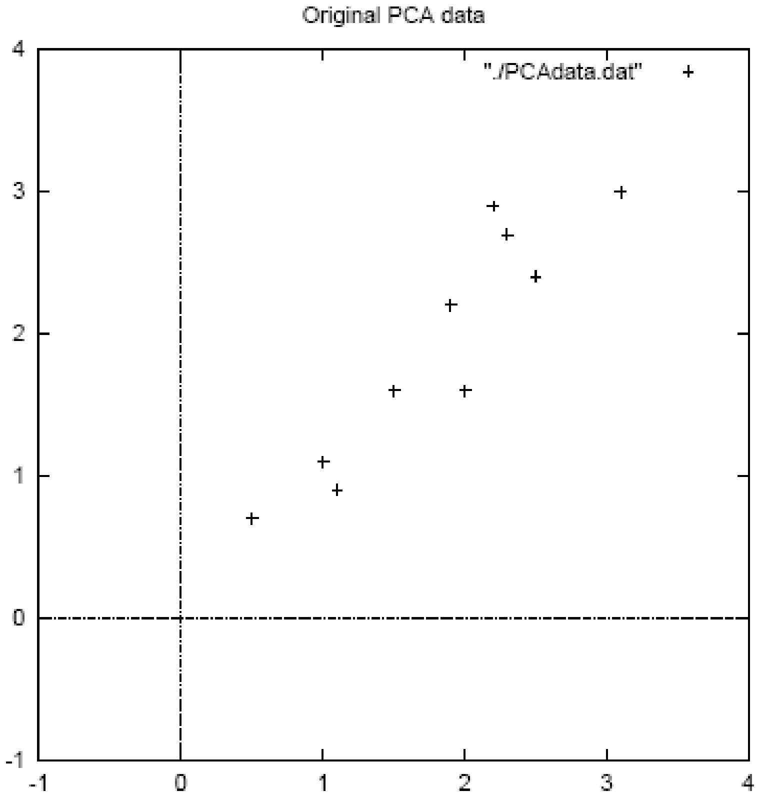
- Threshold alone may not be enough to indicate anomalous behavior because different servers may have different ranges of normality in such features.
- Need some mechanism to learn which pattern is normal and which pattern is abnormal.
- Here comes PCA...

Introduction: PCA

- Principal Component Analysis (PCA) is the technique used for identifying patterns in the data set in form of vectors.
- PCA transforms original data points to the new coordinate axes in order to express their similarities and differences.

Introduction: PCA

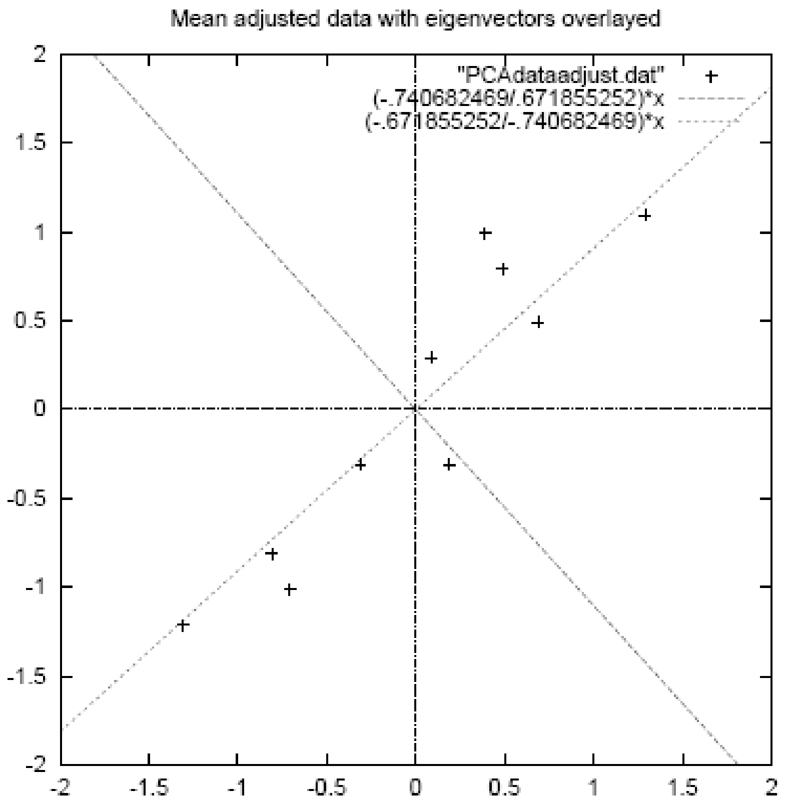
- A quick example



X	Y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2 1	1.6
1	1.1
1.5	1.6
1.1	0.9

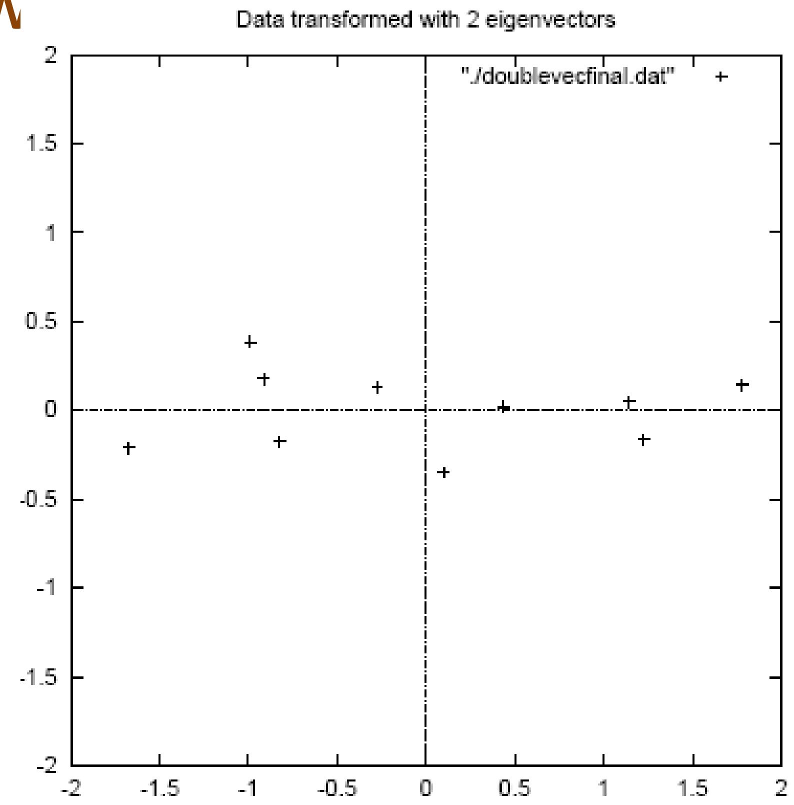
Introduction: PCA

- After some calculations, we have the new PCA axes



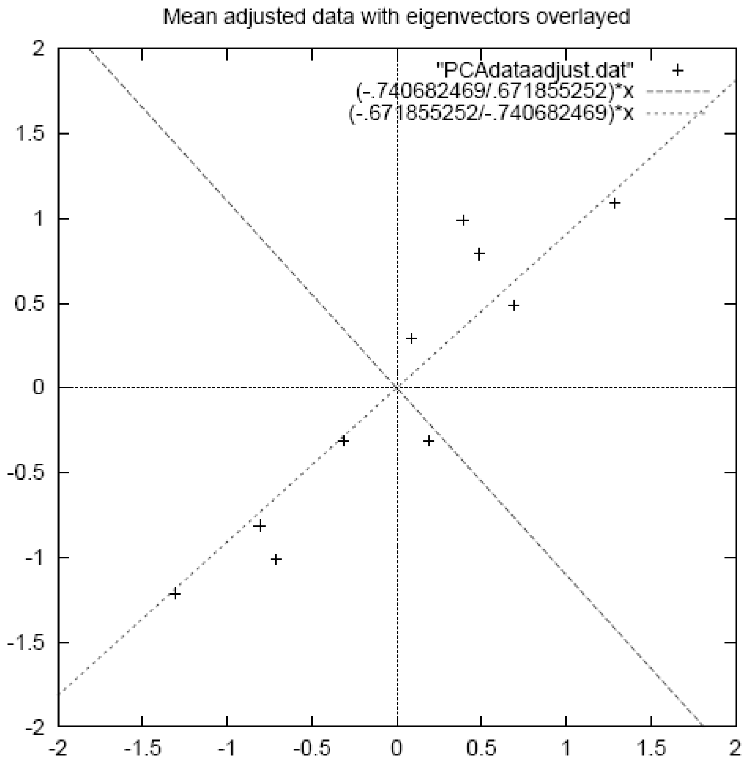
Introduction: PCA

- Map all data points to the new axis to get the new set of coordinates



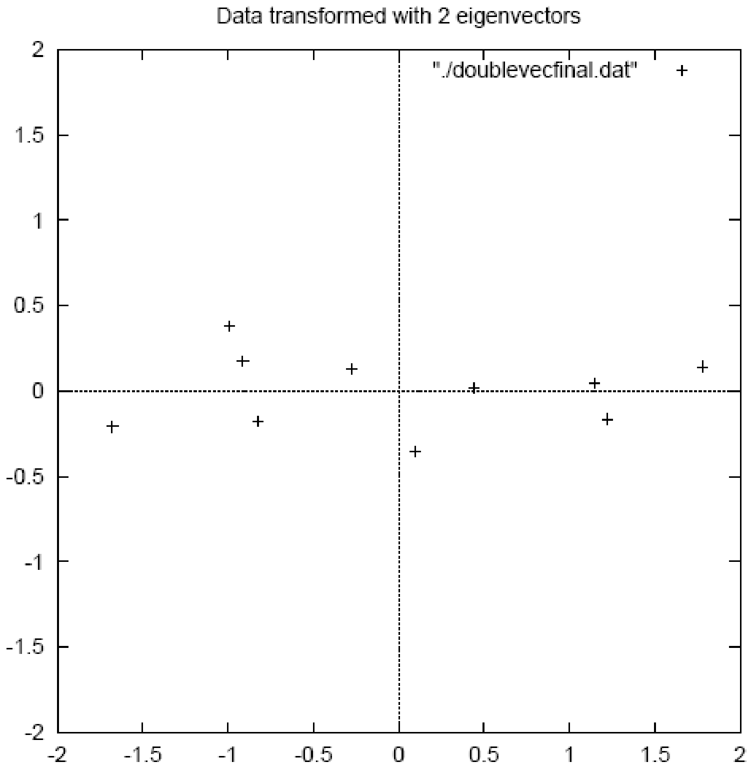
X	Y
-0.827970186	-0.175115307
1.77758033	0.142857227
-0.992197494	0.384374989
-0.274210416	0.130417207
-1.67580142	-0.20949846
-0.912949103	
0.0991094375	
1.14457216	

Introduction: PCA



Original Data

subt



Final Data

Agenda

- Introduction

  * Motivation *Features

  * Anomalies Types *PCA

  * Access log and Error log

Click to edit Master subtitle style

- **Methodology**
- Experiment I: Determining Useful Dimensions
- Experiment II: Anomaly Detection
- Future Works

Methodology

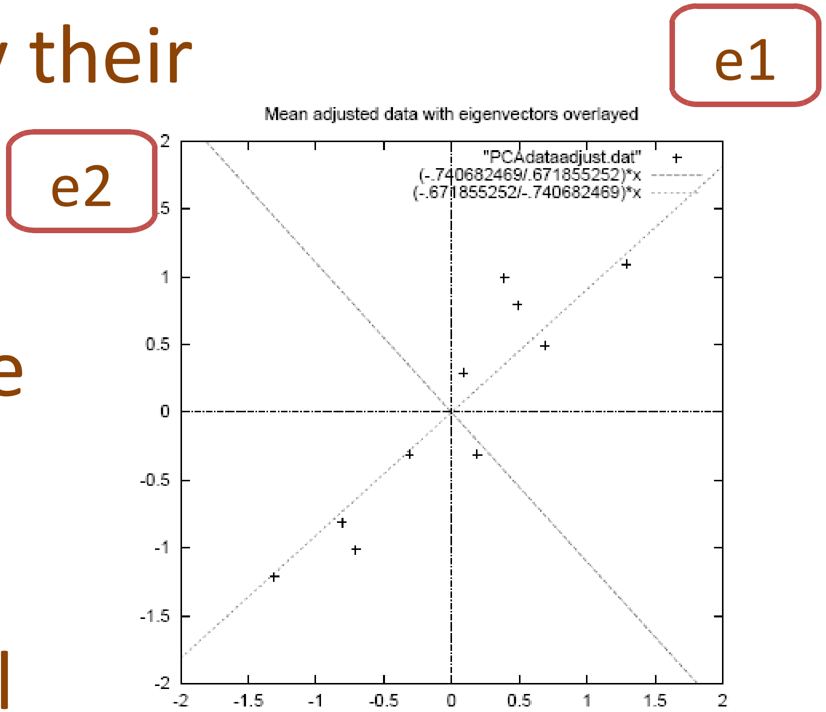
- Prepare the set of data in the form of 10 dimensional feature vectors obtained from the log files.
- Subtract the mean from each vectors
- Prepare the covariance matrix :

$$\begin{bmatrix} \mathrm{Cov}(X_1, X_1) & \mathrm{Cov}(X_1, X_2) & \dots & \mathrm{Cov}(X_1, X_N) \\ \mathrm{Cov}(X_2, X_1) & \mathrm{Cov}(X_2, X_2) & \dots & \mathrm{Cov}(X_2, X_N) \\ \vdots & \vdots & & \vdots \\ \mathrm{Cov}(X_N, X_1) & \mathrm{Cov}(X_N, X_2) & \dots & \mathrm{Cov}(X_N, X_N) \end{bmatrix}$$

$$\mathrm{Cov}(\mathrm{X}, \mathrm{Y}) = \sum \frac{(x_i - \overline{x})(y_i - \overline{y})}{N}$$

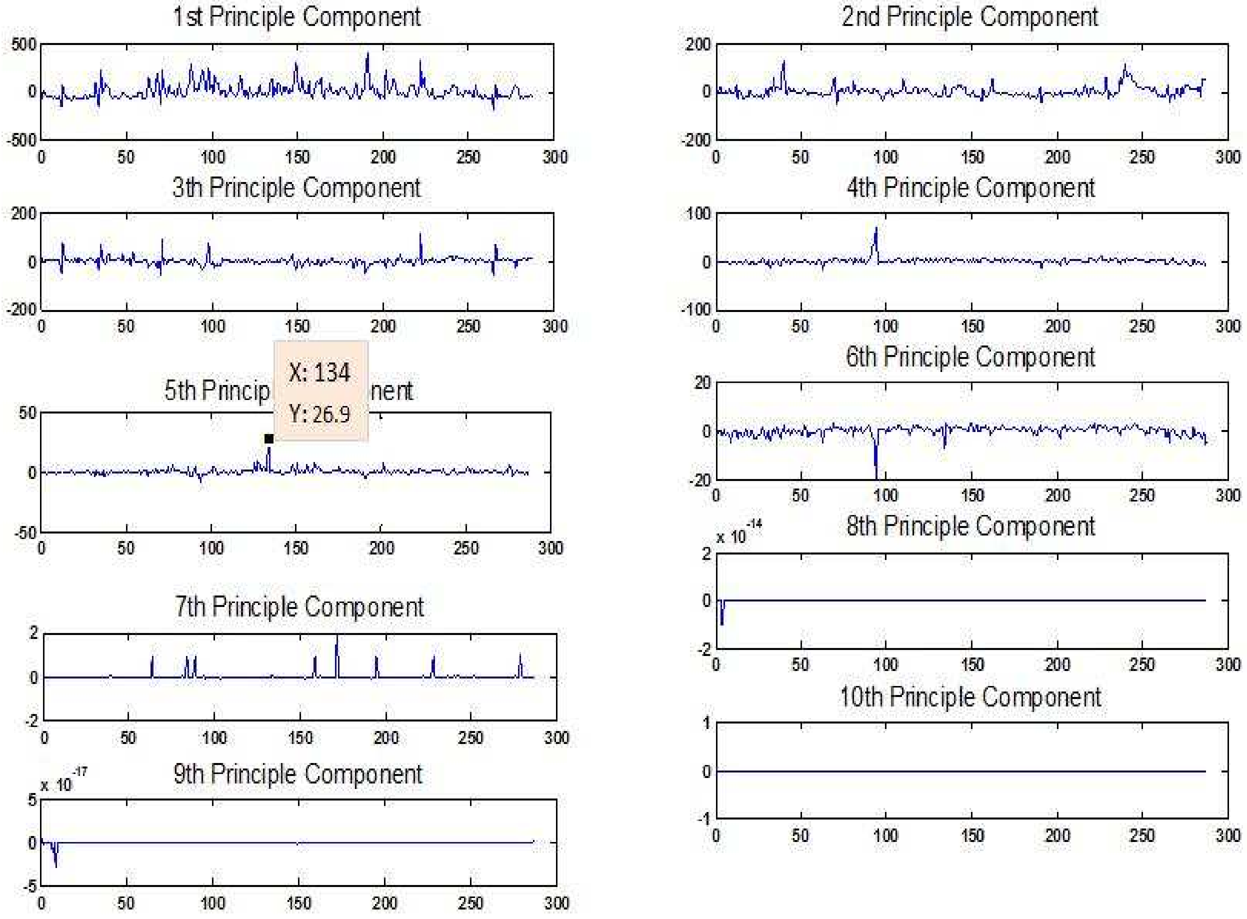
Methodology

- Solve for eigenvectors and eigenvalues of the covaiance matrix.
- Rearrange eigenvectors by their eigenvalues descendingly to obtain the new axes sorted by their significance (e.g. the eigenvector with the highest eigenvalue becomes the first principal axis)



Methodology

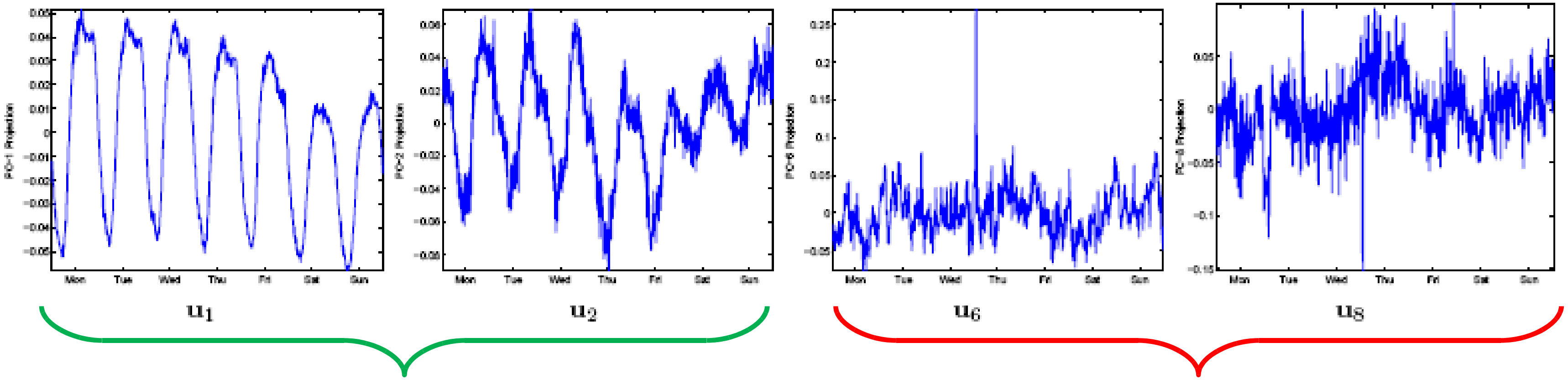
Project data vectors onto PCA axes.



27/9/:

Methodology

- Apply PCA on this set of vectors.
- Select an appropriate separating axis between normal part and anomalous part. The first k-principal axis will form the normal subspace



Methodology

- Let *k* be the dimension of the normal subspace. Pick the time bins whose residual norm

$$\|\tilde{x}\| = \left\| x - \sum_{i=1}^{k} \langle x, e_i \rangle e_i \right\|$$

exceeds the *x* standard deviation threshold. These time bins will be considered to contain anomalies.

Agenda

- Introduction

  * Motivation *Features

  * Anomalies Types *PCA

  * Access log and Error log

Click to edit Master subtitle style

- **Experiment I: Determining Useful Dimensions**
- Experiment II: Anomaly Detection
- Future Works

27/9/10

Experiment I: Anomaly Types

Broken link [HTTP status code 404]

A situation when requested files or requested directories from a web browser cannot be found

Directory Scan [HTTP status code 404, 403]

A situation when a brute-force search is performed with attempts to find valid directories and files in the web server

Click to edit Master subtitle style

Password Cracking [HTTP status code 401]

It is indicated by a series of failed logins within a short period of time by an attacker

Scrip Error [HTTP status code 500]

Unintentional syntax errors of PHP, Perl, or CGI scripts

27/9/10

Experiment I: Data

ApacheLog

- Obtained Apache log files from Traffy Server.
- We use 5 sets of log files from December 23 to December 27, 2009, with avaerage of 2000 visitors per day.
- To simulate anomalous log, we inject four types of events into a dummy webserver. The log files are then merged into the Traffy server.

Click to edit Master subtitle style

27/9/10

Experiment I: Determining Useful Dimensions

Purpose

- To determine dimension of PCA results that would indicate web anomalies

Procedures

*Simulated anomalies in 2 rates and injected each anomaly type at 5 different intervals for each background traffic

*Find 10 features

*Took these features in each background traffic to PCA

Click to edit Master subtitle

Types	Occurring rate 1	Occurring rate 2
Broken Link	60 times per minute For 1 minute	30 times per minute For 1 minute
Password Cracking	60 times per minute For 10 minute	30 times per minute For 10 minute
Scrip Error	60 times per minute For 1 minute	30 times per minute For 1 minute
Directory Scan	805 scans in 35 seconds	805 scans in 35 seconds

Experiment I: Determining Useful Dimensions

Conclusion

Dimensions	Day 1		Day 2		Day 3		Day 4		Day 5	
	Rate 1	Rate 2	Rate 1	Rate 2	Rate 1	Rate 2	Rate 1	Rate 2	Rate 1	Rate 2
Broken Link	5	5	4,5	-	5	5	4,5	5	5	-
Password Cracking	2,4	2,4	3,4	4	2,4	4	2,4	2,4	2,3	4
Scrip Error	5	6	5,6	6	5	6	4,5	6	5	6
Directory Scan	1,2	1,2	1,2,3	1,2,3	1,2	1,2	1,2	1,2	1,2	1,2

Click

- Cannot found

- The first six dimensions seem to be useful for detecting all web anomalies.
- As the attack rate decreases, anomalies tend to show up in lower-variance

27/9/10

Agenda

- Introduction

  * Motivation *Features

  * Anomalies Types *PCA

  * Access log and Error log

Click to edit Master subtitle style

- Experiment I: Determining Useful Dimensions
- **Experiment II: Anomaly Detection**
- Future Works

27/9/10

Experiment II : Anomaly Detection

Purpose

To automatically detect anomalies from the PCA results

Procedures

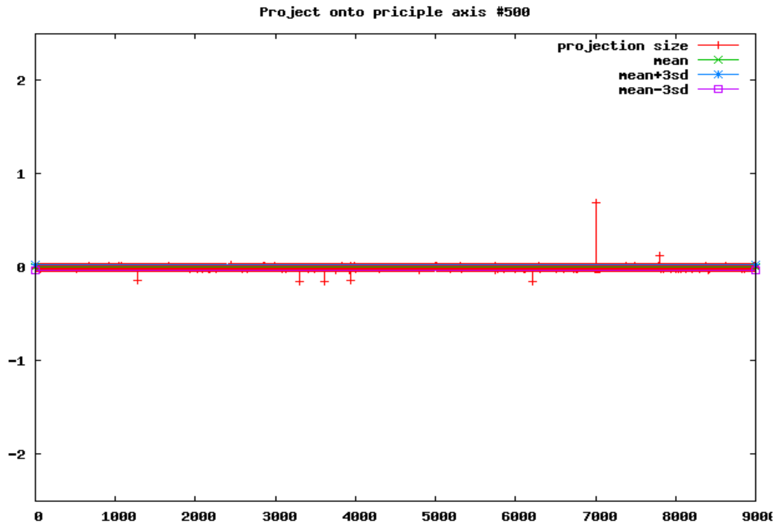
Click to edit Master subtit

*Simulate anomalies at rate2 and injected each anomaly type at 5 different intervals for each log file.

*The web anomalies can be identified within 6 highest-variance dimensions of PCA results.

*Applied thresholds of 4,5 and 6 S.D. to obtain peaks in within those 6 axes

27/9/10

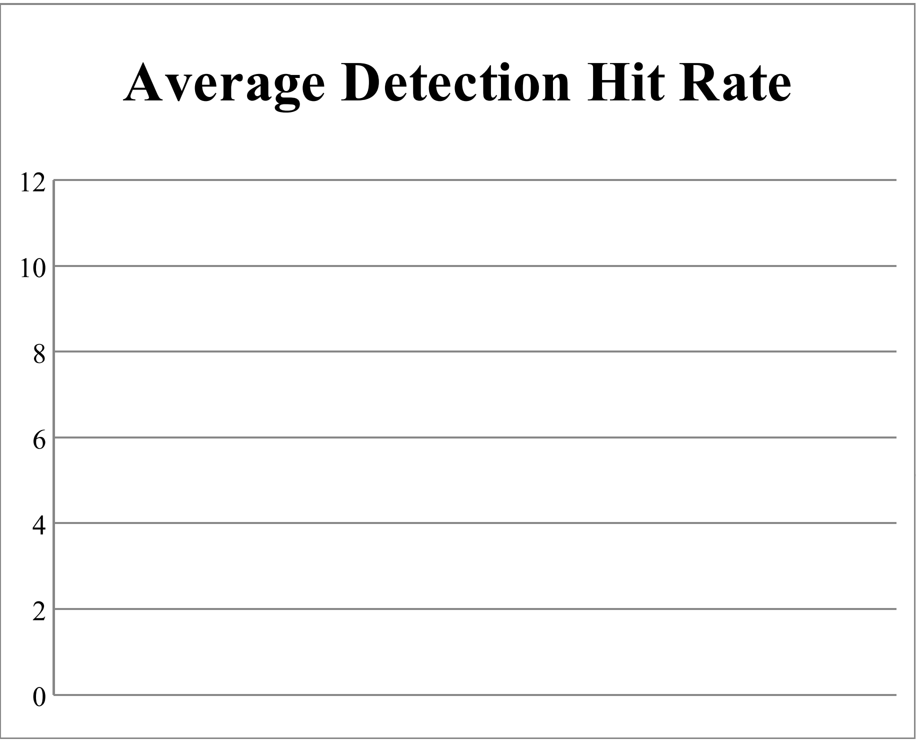


Types of anomalies	Occurring rate 2
Broken Link	30 times per minute For 1 minute
Password Cracking	30 times per minute For 10 minute
Scrip Error	30 times per minute For 1 minute
Directory Scan	805 scans in 35 seconds

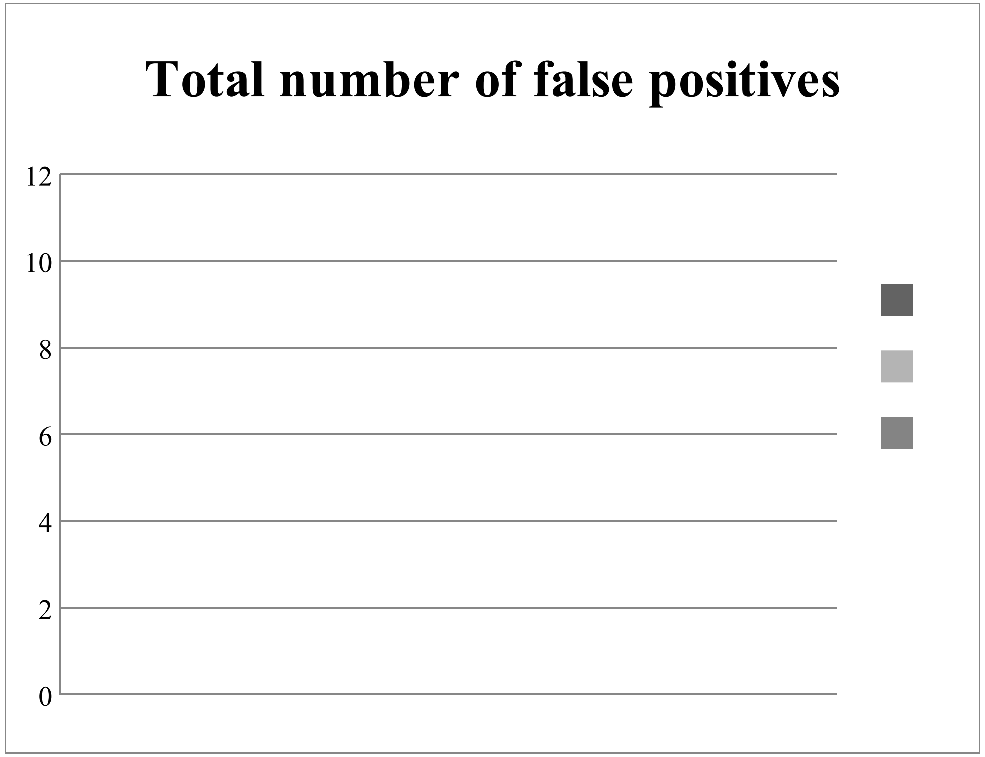
Thresholds
4S.D.
5S.D.
6S.D.

Experiment II : Anomaly Detection

Conclusion



S



- The threshold can detect a directory scan event with the highest rate
- The threshold of 4 S.D yields the best hit rate compared to the other two thresholds

Agenda

- Introduction

  * Motivation *Features

  * Anomalies Types *PCA

  * Access log and Error log

  Click to edit Master subtitle style

- Experiment I: Determining Useful Dimensions
- Experiment II: Anomaly Detection
- **Future Works**

27/9/10

Future Works

- Classification algorithms (Done)
- Extend the set of features, therefore more problem types could be discovered.
- Develop the more robust algorithms for anomaly cutting points.
- Take periodic behaviors into consideration.
- Incremental updating algorithm.

Click to edit Master subtitle style

27/9/10

Any question ?

Click to edit Master subtitle style

Thank you…..

27/9/10