A Statistical Anomaly Detection Approach for Detecting Network Attacks

Roland Kwitt

Salzburg Research &
Univ. of Applied Sciences and Technologies Salzburg

14th December 2004/ 6QM Workshop

Outline

Outline

Connection Between Anomalies and Attacks

- Determining the correspondence between malicious activity and anomalous activity is essential, but not an easy task!
- Based on a generally very huge feature space, a subset of features has to be extracted from which the system can learn a *normal* behavior model
- It is common practice that such models are based on the distributions of the observed features
- Many attacks rely on the ability of an attacker to construct client protocols themselvs. Usually, the target environment is not duplicated carefully

Connection Between Anomalies and Attacks

- Determining the correspondence between malicious activity and anomalous activity is essential, but not an easy task!
- Based on a generally very huge feature space, a subset of features has to be extracted from which the system can learn a *normal* behavior model
- It is common practice that such models are based on the distributions of the observed features
- Many attacks rely on the ability of an attacker to construct client protocols themselvs. Usually, the target environment is not duplicated carefully

Connection Between Anomalies and Attacks

- Determining the correspondence between malicious activity and anomalous activity is essential, but not an easy task!
- Based on a generally very huge feature space, a subset of features has to be extracted from which the system can learn a *normal* behavior model
- It is common practice that such models are based on the distributions of the observed features
- Many attacks rely on the ability of an attacker to construct client protocols themselvs. Usually, the target environment is not duplicated carefully

Connection Between Anomalies and Attacks

- Determining the correspondence between malicious activity and anomalous activity is essential, but not an easy task!
- Based on a generally very huge feature space, a subset of features has to be extracted from which the system can learn a *normal* behavior model
- It is common practice that such models are based on the distributions of the observed features
- Many attacks rely on the ability of an attacker to construct client protocols themselvs. Usually, the target environment is not duplicated carefully

Connection Between Anomalies and Attacks (contd.)

- Network probes and scans are necessarily anomalous since the try to seek information legitimate users already posess
- Already successful executed attacks against a victim host/network often result in so called *response anomalies*
  - Hosts/networks used as traffic amplifiers in DRDoS attacks often show response anomalies
- A thorough description in which way attacks cause anomalies is not possible!
- The power of employing anomaly detection regarding attacks, lies in the fact that you do not need to know anything about an attack!

Connection Between Anomalies and Attacks (contd.)

- Network probes and scans are necessarily anomalous since the try to seek information legitimate users already posess
- Already successful executed attacks against a victim host/network often result in so called *response anomalies*
  - Hosts/networks used as traffic amplifiers in DRDoS attacks often show response anomalies
- A thorough description in which way attacks cause anomalies is not possible!
- The power of employing anomaly detection regarding attacks, lies in the fact that you do not need to know anything about an attack!

Connection Between Anomalies and Attacks (contd.)

- Network probes and scans are necessarily anomalous since the try to seek information legitimate users already posess
- Already successful executed attacks against a victim host/network often result in so called *response anomalies*
  - Hosts/networks used as traffic amplifiers in DRDoS attacks often show response anomalies
- A thorough description in which way attacks cause anomalies is not possible!
- The power of employing anomaly detection regarding attacks, lies in the fact that you do not need to know anything about an attack!

Connection Between Anomalies and Attacks (contd.)

- Network probes and scans are necessarily anomalous since the try to seek information legitimate users already posess
- Already successful executed attacks against a victim host/network often result in so called *response anomalies*
  - Hosts/networks used as traffic amplifiers in DRDoS attacks often show response anomalies
- A thorough description in which way attacks cause anomalies is not possible!
- The power of employing anomaly detection regarding attacks, lies in the fact that you do not need to know anything about an attack!

Outline

The Basics

- While we monitor traffic we observe certain packet header fields (our features) and estimate the parameters of their underlying distribution
- But, how are the header field values distributed ?
- Let a random variable X indicate whether a header field takes on a certain value (denoted by event A, $p := \mathbb{P}(A)$) or not. This simulates a Bernoulli experiment since we only have two outcomes. Thus it follows that

$$X(w) = \begin{cases} 1 & \text{if } w \in A \\ 0 & \text{if } w \notin A \end{cases} \rightsquigarrow X \sim Bernoulli(p) \tag{1}$$

The Basics

- While we monitor traffic we observe certain packet header fields (our features) and estimate the parameters of their underlying distribution
- But, how are the header field values distributed ?
- Let a random variable X indicate whether a header field takes on a certain value (denoted by event A, $p := \mathbb{P}(A)$) or not. This simulates a Bernoulli experiment since we only have two outcomes. Thus it follows that

$$X(w) = \begin{cases} 1 & \text{if } w \in A \\ 0 & \text{if } w \notin A \end{cases} \rightsquigarrow X \sim Bernoulli(p) \tag{1}$$

The Basics

- While we monitor traffic we observe certain packet header fields (our features) and estimate the parameters of their underlying distribution
- But, how are the header field values distributed ?
- Let a random variable X indicate whether a header field takes on a certain value (denoted by event A, $p := \mathbb{P}(A)$) or not. This simulates a Bernoulli experiment since we only have two outcomes. Thus it follows that

$$X(w) = \begin{cases} 1 & \text{if } w \in A \\ 0 & \text{if } w \notin A \end{cases} \rightsquigarrow X \sim Bernoulli(p) \tag{1}$$

The Basics (contd.)

- We repeat the same basic random experiment n times. Let another random variable Y indicate the number of successes: $Y = \#\{i : X_i = 1, i = 1, ..., n\}$. We get

$$Y = \sum_{i=1}^{n} X_i \rightsquigarrow Y \sim B_{n,p} \tag{2}$$

- However, we observe the whole domain D of a header field! Thus, $A_1 \cup \cdots \cup A_k = \Omega$, $k = 1, ..., \#D$.
- The combined probability function of $Y_i, ..., Y_n$, $Y_i \sim B_{n,p_i}$, is given by the multinomial distribution.

$$Z \sim Mn_{n,p_1,...,p_n} \tag{3}$$

The Basics (contd.)

- We repeat the same basic random experiment n times. Let another random variable Y indicate the number of successes: $Y = \#\{i : X_i = 1, i = 1, ..., n\}$. We get

$$Y = \sum_{i=1}^{n} X_i \rightsquigarrow Y \sim B_{n,p} \tag{2}$$

- However, we observe the whole domain D of a header field! Thus, $A_1 \cup \cdots \cup A_k = \Omega$, $k = 1, ..., \#D$.
- The combined probability function of $Y_i, ..., Y_n$, $Y_i \sim B_{n,p_i}$, is given by the multinomial distribution.

$$Z \sim Mn_{n,p_1,...,p_n} \tag{3}$$

The Basics (contd.)

- We repeat the same basic random experiment n times. Let another random variable Y indicate the number of successes: $Y = \#\{i : X_i = 1, i = 1, ..., n\}$. We get

$$Y = \sum_{i=1}^{n} X_i \rightsquigarrow Y \sim B_{n,p} \tag{2}$$

- However, we observe the whole domain D of a header field! Thus, $A_1 \cup \cdots \cup A_k = \Omega$, $k = 1, ..., \#D$.
- The combined probability function of $Y_i, ..., Y_n$, $Y_i \sim B_{n,p_i}$, is given by the multinomial distribution.

$$Z \sim Mn_{n,p_1,...,p_n} \tag{3}$$

Outline

Indroduction

- By assuming that we have enough amomaly-free training traffic, it is possible to estimate the parameters of the header field specific multinomial distribution. Lets call this the *nominal profile*.
- We also define a packet window of the last N packets, which is shifted one position per new packet arrival. Parameters estimation of the window specific multinomial distribution leads to a *current traffic profile*.
- The maximum likelihood estimator $\hat{p}_i$ for the probabilities of a multinomial distribution is $\hat{p}_i = \frac{n_i}{n}$ where n_i denotes the number of occurrences of element i.
- We can now calculate the deviation of the current parameters from the expected parameters for normal traffic.

$$d_i = p_{i\,nominal} - p_{i\,current} \tag{4}$$

Indroduction

- By assuming that we have enough amomaly-free training traffic, it is possible to estimate the parameters of the header field specific multinomial distribution. Lets call this the *nominal profile*.
- We also define a packet window of the last N packets, which is shifted one position per new packet arrival. Parameters estimation of the window specific multinomial distribution leads to a *current traffic profile*.
- The maximum likelihood estimator $\hat{p}_i$ for the probabilities of a multinomial distribution is $\hat{p}_i = \frac{n_i}{n}$ where n_i denotes the number of occurrences of element i.
- We can now calculate the deviation of the current parameters from the expected parameters for normal traffic.

$$d_i = p_{i\,nominal} - p_{i\,current} \tag{4}$$

Indroduction

- By assuming that we have enough amomaly-free training traffic, it is possible to estimate the parameters of the header field specific multinomial distribution. Lets call this the *nominal profile*.
- We also define a packet window of the last N packets, which is shifted one position per new packet arrival. Parameters estimation of the window specific multinomial distribution leads to a *current traffic profile*.
- The maximum likelihood estimator $\hat{p}_i$ for the probabilities of a multinomial distribution is $\hat{p}_i = \frac{n_i}{n}$ where n_i denotes the number of occurrences of element i.
- We can now calculate the deviation of the current parameters from the expected parameters for normal traffic.

$$d_i = p_{i\,nominal} - p_{i\,current} \tag{4}$$

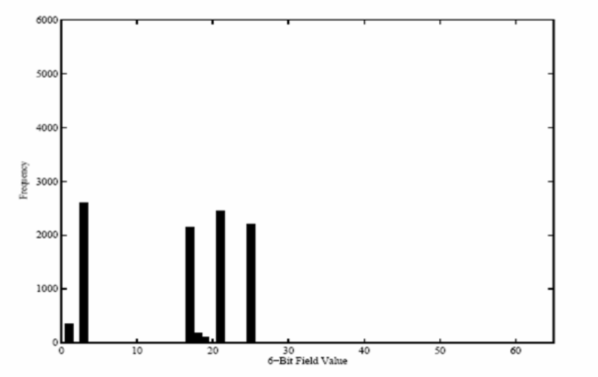
Indroduction

- By assuming that we have enough amomaly-free training traffic, it is possible to estimate the parameters of the header field specific multinomial distribution. Lets call this the *nominal profile*.
- We also define a packet window of the last N packets, which is shifted one position per new packet arrival. Parameters estimation of the window specific multinomial distribution leads to a *current traffic profile*.
- The maximum likelihood estimator $\hat{p}_i$ for the probabilities of a multinomial distribution is $\hat{p}_i = \frac{n_i}{n}$ where n_i denotes the number of occurrences of element i.
- We can now calculate the deviation of the current parameters from the expected parameters for normal traffic.

$$d_i = p_{i\,nominal} - p_{i\,current} \tag{4}$$

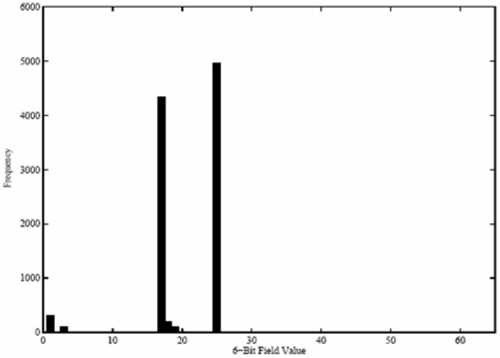
Visualization

- Multinomial distribution of the nominal traffic profile (illustrated as bar chart)



Visualization (contd.)

- Multinomial distribution under an attack (window length equals the length of the nominal profile observation period)



Change Recognition

- Calculate the empirical cumulative distribution function (ECDF) of the oscillations around the expected mean
- Additionally calculate the same ECDF for the last N oscillation values (again *sliding window principle*)
- $\rightsquigarrow$ Two sample Goodness-of-Fit (GoF) tests (Kolmogorov-Smirnov, Chi-Square . . .)
- Problem: Too slow when employed at monitoring systems for high speed links ! Optimal: solution with $O(1)$ complexity
- The difference between the areas under both ECDFs can be calculated iteratively

Change Recognition

- Calculate the empirical cumulative distribution function (ECDF) of the oscillations around the expected mean
- Additionally calculate the same ECDF for the last N oscillation values (again *sliding window principle*)
- $\rightsquigarrow$ Two sample Goodness-of-Fit (GoF) tests (Kolmogorov-Smirnov, Chi-Square . . .)
- Problem: Too slow when employed at monitoring systems for high speed links ! Optimal: solution with $O(1)$ complexity
- The difference between the areas under both ECDFs can be calculated iteratively

Change Recognition

- Calculate the empirical cumulative distribution function (ECDF) of the oscillations around the expected mean
- Additionally calculate the same ECDF for the last N oscillation values (again *sliding window principle*)
- $\rightsquigarrow$ Two sample Goodness-of-Fit (GoF) tests (Kolmogorov-Smirnov, Chi-Square . . .)
- Problem: Too slow when employed at monitoring systems for high speed links ! Optimal: solution with $O(1)$ complexity
- The difference between the areas under both ECDFs can be calculated iteratively

Change Recognition

- Calculate the empirical cumulative distribution function (ECDF) of the oscillations around the expected mean
- Additionally calculate the same ECDF for the last N oscillation values (again *sliding window principle*)
- $\rightsquigarrow$ Two sample Goodness-of-Fit (GoF) tests (Kolmogorov-Smirnov, Chi-Square . . .)
- Problem: Too slow when employed at monitoring systems for high speed links ! Optimal: solution with $O(1)$ complexity
- The difference between the areas under both ECDFs can be calculated iteratively

Change Recognition

- Calculate the empirical cumulative distribution function (ECDF) of the oscillations around the expected mean
- Additionally calculate the same ECDF for the last N oscillation values (again *sliding window principle*)
- $\rightsquigarrow$ Two sample Goodness-of-Fit (GoF) tests (Kolmogorov-Smirnov, Chi-Square . . .)
- Problem: Too slow when employed at monitoring systems for high speed links ! Optimal: solution with $O(1)$ complexity
- The difference between the areas under both ECDFs can be calculated iteratively

More Optimizations

- While estimating the parameters of the multinomial distributions the constraint $\sum_{i=1}^{n} p_i = 1$ must be met.
- A normalization step after each packet arrival would be needed $\rightsquigarrow$ computationally expensive (especially for large domains)
- Due to our iterative *integral* test, only the correct probability for the value that has occured in the current packet is needed.
- Normalization in each step is now obsolete! Result: $O(1)$ complexity of the update routine

More Optimizations

- While estimating the parameters of the multinomial distributions the constraint $\sum_{i=1}^{n} p_i = 1$ must be met.
- A normalization step after each packet arrival would be needed $\rightsquigarrow$ computationally expensive (especially for large domains)
- Due to our iterative *integral* test, only the correct probability for the value that has occured in the current packet is needed.
- Normalization in each step is now obsolete! Result: $O(1)$ complexity of the update routine

More Optimizations

- While estimating the parameters of the multinomial distributions the constraint $\sum_{i=1}^{n} p_i = 1$ must be met.
- A normalization step after each packet arrival would be needed $\rightsquigarrow$ computationally expensive (especially for large domains)
- Due to our iterative *integral* test, only the correct probability for the value that has occured in the current packet is needed.
- Normalization in each step is now obsolete! Result: $O(1)$ complexity of the update routine

More Optimizations

- While estimating the parameters of the multinomial distributions the constraint $\sum_{i=1}^{n} p_i = 1$ must be met.
- A normalization step after each packet arrival would be needed $\rightsquigarrow$ computationally expensive (especially for large domains)
- Due to our iterative *integral* test, only the correct probability for the value that has occured in the current packet is needed.
- Normalization in each step is now obsolete! Result: $O(1)$ complexity of the update routine

Outline

Results

- Evaluation of our approach against the DARPA 1999 Intrusion Detection Data Set
- The analysis algorithms are no longer the performance bottleneck, but the capture routines (even in case of offline analysis)
- Monitored protocols and fields are
  - IP (protocol, ToS, total length)
  - TCP (flags, source port, destination port)
  - UDP (soure port, destination port)
  - ICMP (ICMP type, ICMP code)

Results

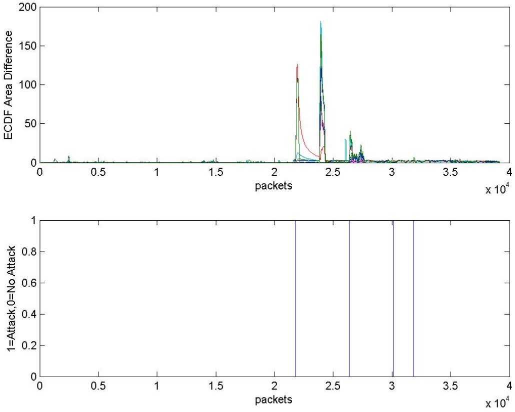
- Evaluation of our approach against the DARPA 1999 Intrusion Detection Data Set
- The analysis algorithms are no longer the performance bottleneck, but the capture routines (even in case of offline analysis)
- Monitored protocols and fields are
  - IP (protocol, ToS, total length)
  - TCP (flags, source port, destination port)
  - UDP (soure port, destination port)
  - ICMP (ICMP type, ICMP code)

Results

- Evaluation of our approach against the DARPA 1999 Intrusion Detection Data Set
- The analysis algorithms are no longer the performance bottleneck, but the capture routines (even in case of offline analysis)
- Monitored protocols and fields are
  - IP (protocol, ToS, total length)
  - TCP (flags, source port, destination port)
  - UDP (soure port, destination port)
  - ICMP (ICMP type, ICMP code)

Visualization

- Analysis of one day of training data (no attacks) and one day of attack data for host *marx*



Further Work

- Increase the subset of observed features
- Include features based on measurements on a higher abstraction level
- Reduce the yet high dimensionality vector to some reasonable one dim. anomaly indicator

Thanks for your attention!