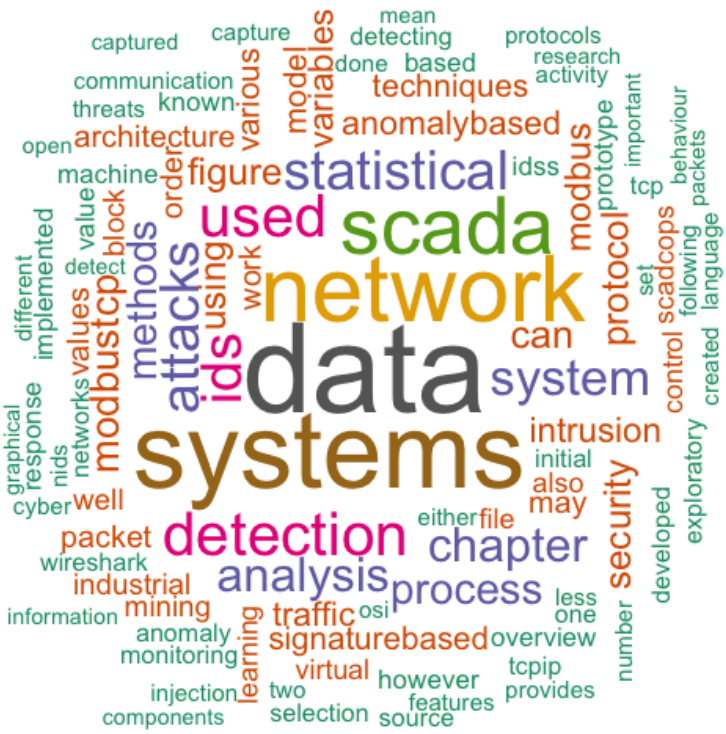
TEST TITLE PAGE

Lisa MALIPHOL

26/08/2015



Diateam: SCAD@COPS
A Hybrid Network Intrusion Detection System

by

Lisa MALIPHOL

A thesis submitted in partial satisfaction of the
requirements for the diploma of the
Masters of Science
in
Computer Science and Decision Systems
in the
Grande École
Télécom Bretagne

Corporate Advisor:
Guillaume Prigent

Academic Advisors:
Professor Yannis Haralambous
Professor Sandrine Vaton

September 2015
Plouzané, FRANCE

Diateam: SCAD@COPS
Un système de détection d'intrusion de réseau hybride

Lisa MALIPHOL

septembre 2015

Plouzané, FRANCE

Contents

Résumé

Mots-clés: détéction d'intrusion réseau, détéction d'intrusion basée des anomalies, systèmes industriels, MODBUS/TCP

Abstract

Keywords: Network Intrusion Detection, Anomaly Intrusion Detection, SCADA Systems, MODBUS/TCP

section 1

```
str(mergedSewDT)
```

```
Classes 'data.table' and 'data.frame':  24945 obs. of  36 variables:
 $ frame.number              : num  2 4 6 8 10 12 14 16 18 20 ...
 $ frame.time_relative       : num  0.18 0.185 0.197 0.208 0.221 ...
 $ frame.time_delta          : num  0.180316 0.000289 0.000254 0.000287 0.000259 ...
 $ frame.len                 : num  66 66 66 66 66 66 66 66 66 66 ...
 $ ip.src                    : Factor w/ 2 levels "","192.168.12.117": 2 2 2 2 2 2 2 2 2 2 ...
 $ eth.src                   : chr  "08:00:27:f9:b1:f1" "08:00:27:f9:b1:f1" "08:00:27:f9:b1:f1" "08:00:
 $ ip.dst                    : Factor w/ 2 levels "","192.168.12.252": 2 2 2 2 2 2 2 2 2 2 ...
 $ eth.dst                   : chr  "00:0f:69:0d:55:cd" "00:0f:69:0d:55:cd" "00:0f:69:0d:55:cd" "00:0f:
 $ mbtcp.modbus.unit_id      : Factor w/ 2 levels "","1": 2 2 2 2 2 2 2 2 2 2 ...
 $ tcp.srcport               : Factor w/ 2 levels "","1043": 2 2 2 2 2 2 2 2 2 2 ...
 $ tcp.dstport               : Factor w/ 2 levels "","502": 2 2 2 2 2 2 2 2 2 2 ...
 $ mbtcp.prot_id             : Factor w/ 2 levels "","0": 2 2 2 2 2 2 2 2 2 2 ...
 $ mbtcp.trans_id            : num  90 91 92 93 94 95 96 97 98 99 ...
 $ mbtcp.len                 : num  6 6 6 6 6 6 6 6 6 6 ...
 $ mbtcp.modbus.func_code    : Factor w/ 2 levels "","4": 2 2 2 2 2 2 2 2 2 2 ...
 $ mbtcp.modbus.word_cnt     : num  1 1 1 1 1 1 1 1 1 1 ...
 $ frame.second              : num  0 0 0 0 0 0 0 0 0 0 ...
 $ mbtcp.modbus.reference_num: Factor w/ 5 levels "","0","1","2",..: 4 2 2 2 2 3 3 5 2 2 ...
 $ resp.frame.number         : num  3 5 7 9 11 13 15 17 19 21 ...
 $ resp.time.rel             : num  0.185 0.197 0.208 0.221 0.233 ...
 $ resp.time.delta           : num  0.00481 0.01157 0.01081 0.01275 0.01174 ...
 $ resp.len                  : num  65 65 65 65 65 65 65 65 65 65 ...
 $ resp.ip.src               : Factor w/ 2 levels "","192.168.12.252": 2 2 2 2 2 2 2 2 2 2 ...
 $ resp.ip.dst               : Factor w/ 2 levels "","192.168.12.117": 2 2 2 2 2 2 2 2 2 2 ...
 $ resp.srcport              : Factor w/ 2 levels "","502": 2 2 2 2 2 2 2 2 2 2 ...
 $ resp.unit_id              : Factor w/ 2 levels "","1": 2 2 2 2 2 2 2 2 2 2 ...
 $ resp.dstport              : Factor w/ 2 levels "","1043": 2 2 2 2 2 2 2 2 2 2 ...
 $ resp.prot_id              : Factor w/ 2 levels "","0": 2 2 2 2 2 2 2 2 2 2 ...
 $ resp.trans_id             : num  90 91 92 93 94 95 96 97 98 99 ...
 $ resp.mbcp.len             : num  5 5 5 5 5 5 5 5 5 5 ...
 $ resp.func.code            : Factor w/ 2 levels "","4": 2 2 2 2 2 2 2 2 2 2 ...
 $ resp.second               : num  0 0 0 0 0 0 0 0 0 0 ...
 $ resp.data                 : Factor w/ 1552 levels "","00:00","00:07",..: 957 31 31 31 31 24 24 605 3
 $ resp.eth.src              : Factor w/ 1 level "00:0f:69:0d:55:cd": 1 1 1 1 1 1 1 1 1 1 ...
 $ resp.eth.dst              : Factor w/ 1 level "08:00:27:f9:b1:f1": 1 1 1 1 1 1 1 1 1 1 ...
 $ d                         : int  4839 112 112 112 112 84 84 3281 112 112 ...
 - attr(*, ".internal.selfref")=<externalptr>
```

section 2

some stuff here

subsection 2.1

subsection stuff here subsection stuff here subsection stuff here

subsection 2.2

subsection stuff here subsection stuff here subsection stuff here

subSubsection 2.2.1

susubsection stuff here subsubsection stuff here subsubsection stuff here

section 3

some stuff here

section 3.1

some stuff here some stuff here some stuff here

section 3.1

some stuff here some stuff here some stuff here

Exploratory Data Analysis

Originally championed by John Tukey[2], Exploratory Data Analysis (EDA) is an initial approach to understanding a data set in order to get a "feel" for the data, to summarizing its essential characteristics and to studying patterns in the data. Moreover, exploratory data analysis frequently incorporates graphical representations beyond using quantitative techniques.

Conducting EDA possibly gives further insight into the form and structure of the data set, in addition to extracting value from it, visualizing it, and just as importantly, in communicating it. After a fairly exhaustive study of the state of the art of IDS and SCADA systems, an initial phase of exploratory data analysis was conducted in order to better understand the data. This section presents a short list of statistical terminology, followed by the exploratory data analysis carried out on the network traffic data captured over the simulated SCADA network.

Statistical Definitions

Mean

The (arithmetic) mean is a measure of central tendency, which is a single value which represents an average of the sample or population. It is calculated by dividing all the observations by the number of observations.

Median

Another measure of central tendency is the median, however, in this case, the median is determined by first ordering the observations by magnitude. Then the median is taken as the value which falls in the middle, or the average of the two middle values in the case of an even number of observations. The median is better suited when there are observations, or outliers, that fall way outside the norm. These are extreme values that differ greatly from other values in the data set.

Variance

The variance is the expected value of the squared differences between the random variables and its mean that is always positive. It gives an indication of how far apart the values are from the mean and each other.

$$var[X] = E[(X - E[X])^2]$$

MODBUS/TCP

The MODBUS/TCP protocol is an open standard and popular network protocol used for ICS devices. It is a messaging protocol located at the application layer that was designed to communicate with PLCs in industrial systems. However, due to the limited resources the PLCs have, it was created to be a simple protocol that provides no security against unauthorized commands or interception of data.[[@Modbus2012]] Figure 5 gives an example architecture for MODBUS TCP communication.

The master initiates a request and the slave sends a response containing either data or error. The common implementations of MODBUS are over Ethernet networks (MODBUS/TCP) or Serial busses (MODBUS/RTU). Both forms of MODBUS contain the packet data unit (PDU), the component consisting of a function code and data.

Attached to the PDU is the application specific addressing and error checking, which together comprise the application data unit (ADU). Specific to MODBUS/TCP, the ADU is encapsulated in the TCP packet. Thereby eliminating the need to include error checking in the MODBUS/TCP layer, it is left out from the MODBUS/TCP ADU. The MODBUS/TCP frame is depicted in Figure 6.

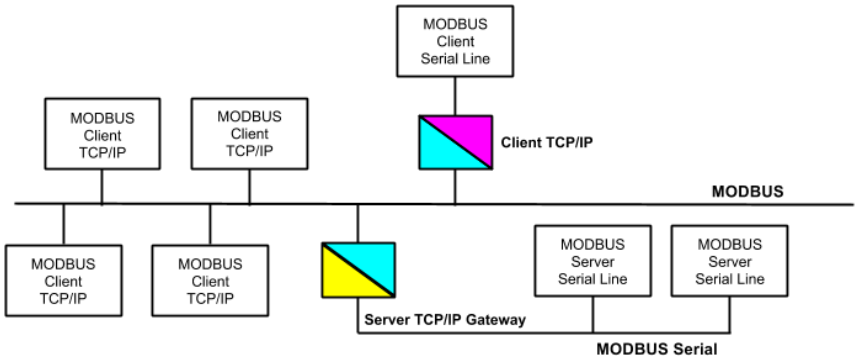


Figure 1: MODBUS TCP/IP Communication Architecture

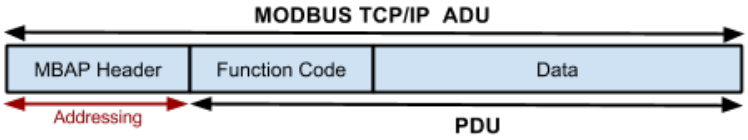


Figure 2: MODBUS/TCP Frame