Filtering real data for IDS benchmarking

MSc Project Plan

Vidar Evenrud Seeberg
vseeberg@netcom.no

Norwegian Information Security Laboratory – NISlab
Department of Computer Science
and Media Technology
Gjøvik University College
P.O. Box 191, 2802 Gjøvik, Norway

9th December, 2005
Abstract

Current IDS evaluation approaches use simulated network traffic as base for the test data sets used in the evaluation. Simulated network traffic lacks the diversities characteristic to a real world network. These diversities may be caused by non-standard implementations of protocols or abnormal protocol behavior, like unfinished three-way TCP handshakes and teardowns. For realistic IDS evaluation, there are a need for test data sets based on real recorded network traffic. Such data sets must also be distributable since a valid test always should be possible to reproduce by other evaluators. Due to legal concerns test data sets based on real recorded traffic must be cleansed for sensitive information. For producing correct IDS metrics the test data sets should also be cleansed for known attacks. This MSc Thesis Project Plan presents a plan for devising a methodology and developing a prototype for cleansing real data. Such a cleansed data set can be used in IDS benchmarking, providing more realistic evaluations. Other benefits are that the test may be reproduced by others and that an IDS' performance may be measured more accurately than today.

Sammendrag

## Revision history

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>20th Oct 2005</td>
<td>First draft, chapter 1 and 2.</td>
</tr>
<tr>
<td>0.2</td>
<td>1st Nov 2005</td>
<td>Second draft, chapter 1 and 2, first draft chapter 3</td>
</tr>
<tr>
<td>0.3</td>
<td>15th Nov 2005</td>
<td>Third draft, all chapters</td>
</tr>
<tr>
<td>0.4</td>
<td>5th Dec 2005</td>
<td>Fourth draft, all chapters, submitted to supervisor</td>
</tr>
<tr>
<td>1.0</td>
<td>9th Dec 2005</td>
<td>Finished</td>
</tr>
</tbody>
</table>

Table 1: Revision history
# Contents

Abstract ................................................................. i  
Sammendrag .............................................................. i  
Revision history ........................................................ ii  
Contents ................................................................. iii  
List of Figures .......................................................... iv  
List of Tables ........................................................... iv  
1 Introduction ......................................................... 1  
  1.1 Topic ................................................................. 1  
  1.2 Problem description ............................................. 1  
  1.3 Justification, motivation and benefits ......................... 1  
  1.4 Research questions ............................................. 2  
2 Previous Work ...................................................... 3  
  2.1 Information revealing subject identity ......................... 3  
  2.2 Known methodologies for anonymizing real network data .... 3  
  2.3 Known methodologies for removing attacks from recorded network data 4  
  2.4 Merging anonymization and attack removal techniques into a new methodology ........................................... 5  
  2.5 IDS metric calculations when using a cleansed test data set ... 5  
3 Claimed contributions ............................................... 6  
4 Choice of methods .................................................. 7  
5 Milestones, deliverables and resources .......................... 9  
  5.1 Milestones and deliverables ..................................... 9  
  5.2 Activities and resources ....................................... 9  
    5.2.1 Literature study ............................................ 9  
    5.2.2 Devise cleansing methodology ............................ 12  
    5.2.3 Prototype development .................................... 12  
    5.2.4 IDS metric calculations ................................. 13  
    5.2.5 Writing report ........................................... 13  
  5.3 Table of Contents for the MSc Thesis ........................ 14  
6 Feasibility study .................................................... 15  
7 Risk analysis .......................................................... 17  
8 Ethical and legal considerations ................................... 19  
Bibliography ............................................................ 20
List of Figures

1 Gantt chart ........................................... 10

List of Tables

1 Revision history ......................................... ii
2 Milestones and deliverables .......................... 11
3 Risks .................................................... 18
1 Introduction

1.1 Topic

Intrusion Detection System (IDS) benchmarking is a necessary task for determining how good IDSs are. Current IDS benchmarking efforts are based on simulated test data sets. Traffic generators are used to generate traffic in which attacks are injected. For more realistic benchmarking, real recorded traffic is needed as a base for the test data set.

The main topic for this thesis is to devise a methodology for filtering prerecorded network traffic in order to generate distributable test data sets for IDS benchmarking. Due to legal concerns, the recorded traffic must be anonymized. For producing correct IDS metrics the filtering process will also contain an attack removal part. A software prototype based on the methodology will be implemented for filtering network data. By comparing the results of applying the cleansed data set to an IDS with applying a simulated data set to the same IDS, it may be possible to say if the cleansed test data set produces fewer false positives.

1.2 Problem description

There are at least two problems with the generation of test data sets using real prerecorded traffic. First, when recording real data, sensitive and identifying data may also be recorded. Due to legal concerns, test data sets containing this type of information can not be distributed. A main property of testing is reproducibility. It should be possible to validate an evaluation through a reevaluation using the same infrastructure, test data and so on. For this to be possible, test data sets must be distributable. Methodologies for anonymizing prerecorded network streams will be investigated in order to produce cleansed test data sets that freely may be distributed.

Second, recorded traffic may also contain attacks. For correct calculations of IDS metrics (such as the probability of false alarms) all known attacks must be removed from the recorded traffic. If the test data set contains no attacks the evaluator may him- or herself insert attacks in a controlled manner. Methodologies for removing attacks from a recorded network stream will therefore be investigated.

1.3 Justification, motivation and benefits

When using simulated traffic generated by network traffic generators in IDS benchmarking, many characteristic properties of network behavior will be left out. Also, different networks have different characteristics. To find out which IDS suits a specific network, different IDSs must be tested with a network traffic pattern characteristic to that specific network.

Traffic generators can not produce traffic that is close enough to the real world for realistic testing. Also, traffic generators are good at producing traffic conforming to protocol specifications. Due to nonstandard implementations of protocols in many applications and other deviating protocol behaviors (like unfinished TCP handshakes and
teardowns), network traffic may contain varying levels of abnormal traffic patterns. For realistic IDS benchmarking, test data sets must be generated based on real traffic from the network type in question. This thesis will describe a methodology and describe a prototype system which can be used to produce clean test data sets for any environment. Network data may be recorded in any type of environment, and then cleansed using the system to be developed. This data set can be used in IDS benchmarking. Such a data set may also be distributed since the data set is anonymized. In this way it may also be possible to build a library of clean data sets to be used for different purposes.

More realistic IDS benchmarking will be helpful for vendors producing IDSs, for customers buying them and for administrators tuning them. Vendors will get better information of what their products are good at and not so good at. This will help vendors improve their products and even enable the vendors to tune the IDSs for specific network types. Customers will easier be able to determine which IDS to purchase, and administrators will have more detailed information when tuning the performance of the systems.

1.4 Research questions

The following questions will be investigated:

1. Which methodologies can anonymize data from a real network?
2. Which methodologies can remove attacks from recorded network traffic?
3. Can methodologies for anonymization of network data and attack removal be merged into a methodology for cleansing real network data?
4. Will correct calculations of IDS metrics be possible using cleansed network traffic as base for IDS benchmarking?
2 Previous Work

The research questions in the preceding chapter states the areas for this thesis’ research. In addition to the research questions it is important to know what information may reveal a subject’s identity in regard to network traffic. This is not a research question in that this information is already known. However, since this area forms an essential background to the anonymizing part of this thesis, this topic is included in this chapter.

2.1 Information revealing subject identity

The problem with revealing identity regarding distribution of test data sets is clearly stated in [1]. [2, 3] provide good descriptions of what type of information to be considered sensitive. In this context, sensitive information means information revealing a subject’s identity. [2] states that information subject to anonymization fall into two categories: identities, including identity of users, hosts and data, and confidential attributes like passwords and specifics of sensitive user activity. [3] gives a thorough list of fields in IP, TCP, UDP and application layer headers that should be considered in an anonymization methodology. Both [3] and [2] also gives examples of how different types of information may be correlated to reveal a subject’s identity.

2.2 Known methodologies for anonymizing real network data

Earlier anonymization work have concentrated on anonymizing packet headers. [4] have presented a method based on cryptography for IP address anonymization while still preserving a common prefix of the address space. [5] have presented a scheme for packet trace anonymization where the results are stored in a compressed format. Most such approaches have completely removed the payload.[2].

More recent approaches have also taken the payload into consideration. [2] describes an anonymizer developed as an extension to the Bro Intrusion Detection System [6]. The paper describes several techniques including:

- constant substitution, e.g. any password may be substituted with the string `<password>`
- sequential numbering, e.g. file names may be substituted, like `<file1>`, `<file2>` and so on
- hashing, payload is replaced with its HMAC-MD5 hash value
- prefix-preserving mapping, e.g. the first part of IP-addresses or directory components of file names are hashed, indicating common values
- adding random noise to numeric values

Another interesting approach is the network dump data displayer and editor (NetDude)[7]. NetDude is a framework for packet trace manipulation. The implemen-
tation of NetDude described in [8] includes an API\(^1\) the author claims could be used to develop an anonymizer plugin to NetDude.

One of the main goals of this thesis is to describe how identifying information can be removed from recorded traffic in a way that

1. no information needed for intrusion detection are removed and
2. assurance are attained that no private and sensitive information remain?

There will always be a possibility that information needed for intrusion detection are removed. This is especially true regarding unknown attacks. Anonymization may therefore render detection of such attacks even harder. [2] and [3] have made significant contributions to this topic in regard to known attacks.

Techniques used in data mining may also be used for anonymization. [9] gives a good overview over terminology in this field. [10] mentions several techniques for privacy preserving data mining, including heuristic-based, cryptography-based and reconstruction-based techniques. Heuristic-based techniques may use perturbation (adding noise) or blocking (substitution). Cryptographic techniques are mostly used in distributed data mining as techniques to partition data. Reconstruction-based techniques use reconstruction of objects (e.g. aggregation) in the data mining approach. For this thesis the heuristic- and reconstruction-based algorithms may be interesting starting points.

A problem in regard to anonymization is to what extent assurance that no sensitive information remain in the data set can be made. It is not possible to be completely sure that all identifying information are anonymized without doing manual inspection. [2] approaches this using a fail-safe filter-in method where everything that need to be in the clear is explicitly stated and everything else is anonymized. Since there always will be uncertainty about remaining identity information, a methodology for anonymization must therefore include techniques for manual inspection. This thesis' anonymization methodology will try to improve existing methodologies in such a way that manual inspection is kept at a minimum.

2.3 Known methodologies for removing attacks from recorded network data

Little work has been done in this area. [1] recognizes this topic as a research topic. Such a methodology may use techniques used in IDS products like the open source IDSs Snort [11, 12] and Bro [13, 14, 15, 6].

Like for anonymizing identity information, one can never be sure that attacks remain after the removal process. This is because there are no means for removing unknown attacks. As [1] states, this may not be a big concern in regard to hit rate IDS testing. Remaining attacks may have impact on false positive testing, though.

Techniques used in data mining may also be used to discover attacks in prerecorded traffic. [16] describes an approach for using data mining programs to make IDSs learn rules that capture the behavior of intrusions and normal activities. This includes techniques for detecting both known and unknown attacks. The Audit Data Analysis and Mining (ADAM) system[17] is a testbed for using data mining techniques in intrusion detection. ADAM uses a combination of association rules mining and classification to

\(^1\)Application Programming Interface
discover attacks in \texttt{tcpdump}[18] audit trails.

2.4 Merging anonymization and attack removal techniques into a new methodology

Little work has been done in this area. The new methodology may use the Bro IDS as a starting point. This IDS can discover attacks and it also has an extension for anonymizing network data. The methodology can try to merge these approaches using Bro as a platform for the methodology. Also, these modules may be used to build a cleansing prototype extension to Bro.

2.5 IDS metric calculations when using a cleansed test data set

There have been very little work on the topic of getting accurate IDS metric calculations when using a cleansed test data set. This is due to the lack of methodologies for cleansing data sets. As mentioned, [1] states that remaining attacks will have an impact on the false positive rate. The question of how anonymized traffic may influence the false positive rate remains unanswered.
3 Claimed contributions

In this thesis an attempt will be made to combine anonymization techniques with attack removal methodologies.

The contribution of this thesis will be to:

1. describe a methodology for combining anonymization and attack removal in order to provide clean test data sets for IDS testing, and to
2. develop a prototype taking recorded network data as input and producing a cleansed data set as output.

This type of methodology and system will be of great importance to IDS evaluation. The testing procedure can be controlled more thoroughly and the test results will be more reliable than today.
4 Choice of methods

This research project is not limited to using just one research method. The research will be of quantitative nature. One common methodology for all the research questions is literature study, which in quantitative research is common at the beginning of a research process [19]. The following will describe which research methodologies should be appropriate for each of the research questions. For a thorough description of the methodologies, see [20] and [19].

Information revealing a subject’s identity This part of the project requires literature study. The part(s) of a network package revealing information about a subject’s identity must be identified. This may be a single header field, multiple header fields, payload or payload in combination with one or several header fields. As mentioned in Chapter 2.1 there are some papers about this topic. Since this is an area concerning legal topics, Norwegian law [21] must also be consulted.

Methodologies for anonymizing data from a real network This topic is closely related to the former topic so literature study applies here as well. In addition to studying articles about the topic, the source code for some open source implementations of anonymizers, like the anonymizing extension to Bro [2] will be studied. Deduction based on premises found in literature and source code will be made in order to devise the methodology. One of the main goals is to develop a prototype of a cleansing system which among else anonymizes network traffic. Prototyping is therefore a research methodology to be used in regard to this topic. Experiments for evaluating the prototype will also be conducted.

Methodologies for removing attacks from recorded network traffic In this topic the same research methodologies as the former topic will be utilized. Literature study of attack detection and removal, and source code study of open source IDSs like Bro and Snort will be useful. Deduction based on premises found in literature and source code will also here be made in order to devise the methodology. In addition prototyping will be used in that the prototype will include an attack removal part.

Merging anonymization and attack removal techniques into a new methodology This topic will use knowledge derived from the two former topics. This implies that much the same methodologies will be used. Deduction based on premises derived from the two former topics will be the main method. Prototyping a system for attack removal in combination with anonymization, and conducting experiments with the prototype, will be done in order to evaluate the cleansing methodology devised.

Correct calculations of IDS metrics using cleansed network traffic. Also for this topic, literature study applies. There are several books about IDSs that also include how IDSs are measured, among these is [22]. Statistical methods may also
be applied. To measure the effectiveness using the prototype, experiments have to be conducted using cleansed data in some experiments and artificially generated data in other experiments.
5 Milestones, deliverables and resources

Figure 1 shows a preliminary Gantt chart for the project. The project is split into parts according to the main tasks to be handled. Some of the tasks will proceed simultaneously. An example is the task of devising a methodology. This task may be influenced by both literature study and implementation. This is also the reason for splitting up in tasks/parts and not in distinct phases. During the first part, literature study, theoretical answers to the research questions will be elaborated. The next part is devoted to devising a methodology for cleansing real data. The third part is developing the prototype. After that the IDS metric calculations will be done. The report will be worked on continuously during all the parts.

5.1 Milestones and deliverables

Table 2 lists the milestones and deliverables for this project. In Figure 1 the milestones are green diamonds and the intermediate deliverables are red diamonds. Each task is followed by a milestone. The first milestone will be signing the contract at project approval. The literature study and the devised methodology will be followed by milestones consisting of preliminary chapters in the report. The milestone of the development phase will be a prototype with source code. This milestone will also be accompanied by a preliminary chapter in the report. The metric calculations will also be followed by a milestone consisting of a preliminary chapter in the report. 1st June is the milestone for delivering the preliminary thesis to the supervisor and the opponent. All these milestones will be considered drafts until the report will be finished and delivered 30th June.

Some of the milestones will be built from intermediate deliverables. The "Previous work" milestone will be built from four intermediate deliverables. The "Methodology" milestone will be built from three intermediate deliverables. And finally, two intermediate deliverables will lead to the prototype and its documentation.

5.2 Activities and resources

Each task is divided into several activities. As depicted in Figure 1 some activities will be executed in parallel. As an example, the literature study part will share its time with devising methodology, implementing prototype and writing report. This also means that estimating exact hours for each activity is difficult. For this project a schedule with overlapping tasks seems more appropriate than a schedule with distinct phases. For example will the methodology be influenced by the prototype development and literature study. The report will also be written continuously during the project.

The number of days noted in this chapter is working days. This means that weekends and national holidays are not included. Each working day is set to 8 hours.

5.2.1 Literature study

This activity already started autumn 2005 in preparations of this paper (MSc Project Plan). The literature study will form the basis for the rest of the project, in particular
Filtering real data for IDS benchmarking

Figure 1: Gantt chart

devising the methodology. Some time will also be spent building competence in programming, especially in C++. This period will also contain some readings of source code of existing anonymizers in order to see how this has been solved before. Bro and NetDudue may be used as subjects in the research. Some time studying these will also be spent.

**Time needed**

Literature study is estimated to last 40 working days, equal to 320 working hours. Note that some of this time will be shared with devising the methodology, developing the prototype and writing the report. The actual time spent on literature study will therefore be less than 320 hours.
<table>
<thead>
<tr>
<th>Date</th>
<th>Milestone / Delivery</th>
<th>Type</th>
<th>In project part / Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>13(^{th}) Jan</td>
<td>Identity-revealing information</td>
<td>D</td>
<td>In literature study / Part of MSc Thesis &quot;Previous Work&quot; chapter</td>
</tr>
<tr>
<td>20(^{th}) Jan</td>
<td>Sign contract</td>
<td>M</td>
<td>Project approval</td>
</tr>
<tr>
<td>27(^{th}) Jan</td>
<td>Anonymization technologies</td>
<td>D</td>
<td>In literature study / Part of MSc Thesis &quot;Previous Work&quot; chapter</td>
</tr>
<tr>
<td>10(^{th}) Feb</td>
<td>Attack detection&amp;removal</td>
<td>D</td>
<td>In literature study / Part of MSc Thesis &quot;Previous Work&quot; chapter</td>
</tr>
<tr>
<td>10(^{th}) Feb</td>
<td>Anonymizer part</td>
<td>D</td>
<td>In Cleansing methodology / Part of MSc Thesis &quot;Cleansing Methodology&quot; chapter</td>
</tr>
<tr>
<td>24(^{th}) Feb</td>
<td>IDS Metrics</td>
<td>D</td>
<td>In literature study / Part of MSc Thesis &quot;Previous Work&quot; chapter</td>
</tr>
<tr>
<td>24(^{th}) Feb</td>
<td>Previous Work</td>
<td>M</td>
<td>Preliminary chapter in MSc Thesis</td>
</tr>
<tr>
<td>24(^{th}) Feb</td>
<td>Architecture baseline</td>
<td>D</td>
<td>In Prototype development</td>
</tr>
<tr>
<td>3(^{rd}) Mar</td>
<td>Attack removal part</td>
<td>D</td>
<td>In Cleansing methodology / Part of MSc Thesis &quot;Cleansing Methodology&quot; chapter</td>
</tr>
<tr>
<td>17(^{th}) Mar</td>
<td>Prototype Version 1</td>
<td>D</td>
<td>In Prototype development</td>
</tr>
<tr>
<td>24(^{th}) Mar</td>
<td>Combining approaches</td>
<td>D</td>
<td>In Cleansing methodology / Part of MSc Thesis &quot;Cleansing Methodology&quot; chapter</td>
</tr>
<tr>
<td>24(^{th}) Mar</td>
<td>Methodology</td>
<td>M</td>
<td>Preliminary chapter in MSc Thesis</td>
</tr>
<tr>
<td>12(^{th}) Apr</td>
<td>Prototype delivery</td>
<td>M</td>
<td>In Prototype development</td>
</tr>
<tr>
<td>28(^{th}) Apr</td>
<td>Prototype documentation</td>
<td>M</td>
<td>Preliminary chapter in MSc Thesis</td>
</tr>
<tr>
<td>12(^{th}) May</td>
<td>Results metric analysis</td>
<td>M</td>
<td>In Metric testing and analysis / Preliminary chapter in MSc Thesis</td>
</tr>
<tr>
<td>1(^{st}) Jun</td>
<td>Submit report to supervisor and opponent</td>
<td>M</td>
<td>Preliminary MSc Thesis</td>
</tr>
<tr>
<td>30(^{th}) Jun</td>
<td>MSc Thesis Delivery</td>
<td>M</td>
<td>Finished</td>
</tr>
</tbody>
</table>

\(^{a}\) D=Intermediate Delivery, M=Milestone

Table 2: Milestones and deliverables


**Necessary equipment**

This activity requires the necessary papers for reading. A computer and a printer to print the papers and source code are also needed. Trying out Bro and NetDude requires a computer with network connectivity. Books teaching C++ and Java programming languages are needed for acquiring coding competence. There are also tutorials and references for these languages on the Internett, which should be accessible. This activity will result in a "Previous work" chapter in the Thesis. The writings will be done in L\LaTeX\ and presentations will be made in OpenOffice Impress and converted to .pdf-files. These requirements are already met by Gjøvik University College through normal student access and by myself.

**Contributions from others**

Professor Slobodan Petrović, is supervising this MSc Thesis. He will be consulted regarding relevant papers, Bro and NetDude. A subscription to the mailing list for the Bro project have been made to get advice on questions concerning Bro. Contact with one of NetDude’s contributors, Christian Kreibich, is also established.

5.2.2 Devise cleansing methodology

In this part of the project the methodologies and algorithms for attack removal and anonymization will be chosen and merged into a new methodology for cleansing data. This theoretical methodology will form the basis for the prototype development. However, the methodology will continue to emerge in parallel with the prototype since these tasks may influence on each other.

**Time needed**

Devising the cleansing methodology is estimated to last 45 working days, equal to 360 working hours. Note that some of this time will be shared with literature study, developing the prototype and writing the report. The actual time spent on devising the methodology will therefore be less than 360 hours.

**Necessary equipment**

This activity does not require any other equipment than the papers and the computer mentioned in Chapter 5.2.1.

**Contributions from others**

The supervisor of this project will be consulted.

5.2.3 Prototype development

The prototype development will contain several subactivities. A development methodology, like Unified Process or Waterfall, is by this time of writing not chosen. At this point only the need for the subactivities architecture, development and test are stated. Other subactivities may be added or may replace these. The architecture delivery will be thoroughly audited so the basis for the prototype is sound. Version 1 of the prototype will perform simple tasks with just a subset of functionality. The full version prototype will meet the requirements of proving the concept of the methodology.

**Time needed**

Prototype development is estimated to last 48 working days, equal to 384 working hours. Note that some of this time will be shared with literature study, devising metho-
dology and writing the report. The actual time spent on prototype development will therefore be less than 384 hours.

**Necessary equipment**

This activity requires a computer with a C++ or Java Integrated Developer Environment like Eclipse or Anjuta, the last one being only for C++. Books and reference for C++ or Java must also be available. A test data set consisting of recorded network traffic must be available for testing the prototype. Most of these requirements are met at the time of writing. For the test data set, the infrastructure and permission to record (see Chapter 8) is in place, although the recording has not yet been done. The recording will be done using a connection to Uninet from Sandbox Laboratory, using one of the workstations in the lab. The workstation will be equipped with a Linux distribution (Fedora Core 4 or Ubuntu 5.10) or FreeBSD 6.0.

**Contributions from others**

The supervisor of this project will be consulted. The Bro and the NetDude developers will probably also be consulted.

### 5.2.4 IDS metric calculations

This activity follows established methodologies. After cleansing some recorded network data using the prototype, tests will be made for determining if a cleansed test data set will give a lower false positive rate than an artificially generated test data set. The cleansed test data set will first be replayed to see how an IDS responds to the traffic. Afterwards artificially generated traffic will be sent into the network to see how the same IDS responds to this traffic. Depending on how advanced the traffic generators are, the traffic may be varied simulating different types of networks.

**Time needed**

IDS metric calculations is estimated to last 18 working days, equal to 144 working hours. Note that some of this time will be shared with report writing. The actual time spent on calculations will therefore be less than 144 hours.

**Necessary equipment**

This activity requires two computers, one equipped with an IDS and the other with replaying software like tcpdump and a traffic generator like Distributed Internet Traffic Generator (D-ITG)[24]. A network connection for recording traffic is also required. As IDS, Bro or Snort may be used. Network connectivity to Uninet is already been permitted as noted in Chapter 8. The recording will be made in Sandbox Laboratory using the same equipment as mentioned in Chapter 5.2.3. Books regarding IDS metrics must be available. All these requirements are met at the time of writing.

**Contributions from others**

The supervisor of this project will be consulted.

### 5.2.5 Writing report

Writing the report is an ongoing activity starting already in the third week of the project. Probably it will start earlier in the form of notes taken during the two first weeks of literature study. Some periods will be more active than others in regard to writing the report. The last part of each main task will probably be more focused towards report writing since each task will contribute to the project with a milestone
delivery. The milestones will mostly be formed as chapters in the report. The prototype will be presented as a description with the source code listed in an appendix.

**Time needed**

Writing report is estimated to last 113 working days, equal to 904 working hours. Note that some of this time will be shared with the other activities. The actual time spent on writing the report will therefore be less than 904 hours.

**Necessary equipment**

This activity requires a computer with a \LaTeX\ environment. This is already provided.

**Contributions from others**

The supervisor of this project will be consulted.

### 5.3 Table of Contents for the MSc Thesis

This is the preliminary Table of Contents for the MSc Thesis:

- Abstract
- Acknowledgment
- 1. Introduction
  - 1.1 Topic
  - 1.2 Problem description
  - 1.3 Justification, motivation and benefits
  - 1.4 Research questions
- 2. Previous work
- 3. Cleansing methodology
  - 3.1 Anonymization part
  - 3.1 Attack removal part
  - 3.1 Combining anonymization and attack removal
- 4. Cleansing prototype implementation
- 5. Testing using cleansed and simulated data sets
  - 5.1 Test environment
  - 5.2 Testing using a cleansed data set
  - 5.3 Testing using simulated traffic
  - 5.4 Analysis of test results
- 6. Discussion
- 7. Conclusion
- 8. Future work
- Appendices
- References
6 Feasibility study

The methodology for cleansing real data should be possible to devise in the available time frame. Some methodologies exist for anonymization of network data. The attack removal part may build on existing IDS technology. The problem could be to extract the information needed from different sources, like literature and source code. Although not fully analyzed by now, one could think that the two processes work sequentially, doing attack removal first and anonymization next. There are though other problems that might occur. The most important is that the anonymization process could change the test data set in such a way that it would affect an IDS’ functionality at an evaluation. Current anonymization techniques do not take into account that the resulting data is going to be used in IDS evaluation, a reason that motivates this research. This uncertainty encourages a thorough analysis of which type of information to anonymize and in what way.

Another problem is the uncertainty that attacks and identifying information may remain after the cleansing process. This is a typical Turing-stop problem. Because of this, one can never be sure that no attacks and/or no identifying information remain. That is why the cleansing process also needs to have a part of manual inspection. This is especially important for the anonymization process due to the legal aspects. Remaining attacks is not that problematic, only causing some uncertainty in regard to the IDS metric calculations. Uncertainty regarding remaining information after anonymization is recognized as inherent problems in the IDS testing community stated e.g. in [1]. The same paper also states the need for research in cleansed test data sets, and is one of the motivations for this thesis.

In the development part, a prototype will be developed. This is not a fully functional software product. A prototype is built to demonstrate how this type of software might work. It could be further developed into a fully functional product or it could be discarded. If discarded, the knowledge derived from developing and using the prototype could be used in building a functional model from scratch or devising a theoretical model. An anonymizer extension to the open source IDS Bro has been described in [2]. Bro and its extensions are coded in C++. If deciding to use Bro as a subject for research, C++ must be learned. This should not be too difficult. Problems handling Bro’s API may be expected, although this is not considered a major problem.

Using an IDS as subject for research, its ability to detect attacks may be relied upon in regard to the attack removal module. The downside of this approach is that the particular IDS may be favored at an IDS evaluation when a test data set cleansed with the same IDS is used. This speaks for the choice of developing a standalone prototype. If deciding that the prototype should be standalone this might be a huge task implementing anonymization and attack removal from scratch. Implementing analyzers for attack removal and anonymizers for all protocols may not be feasible. Instead only some protocols would be prioritized.

Doing calculations based on testing with a cleansed test data set in comparison to a data set based on recorded traffic should not pose any significant problems. How
the generated data set should be generated, and how similar in type of network traffic the generated data set will be in comparison to the cleansed data set is a question to elaborate, though.
7 Risk analysis

The risk of failing in the process of devising a methodology for cleansing real data is not significant. There are some literature available on the subjects and the different subtopics are relatively clear and concise. The combination of anonymizing and attack removal in one methodology should be a low to medium risk task. Should any problems turn up, the scope of the methodology may be limited.

In regard to the prototype development the risk is greater. As mentioned in Chapter 6 the open source IDS Bro or the NetDude software might be used as subject for the research. Using Bro will save a lot of time in that algorithms for detection and anonymization are present. There may be problems in approaching the API of Bro and NetDude, but contact with the developers of these products is established. Therefore, these problems should not be significant and the risk should be considered low to medium. To counter the risk, Bro and NetDude’s approachability should be tested at an early stage. If the results show problems in interfacing them, an implementation of a standalone system for cleansing real network data may be considered. Another approach might be to describe a prototype based on Bro or NetDude in words.

If the standalone prototype approach is chosen, two risks must be handled. First, all algorithms must be implemented personally, including equivalents to those that might have been taken advantage of as implemented in Bro, if that approach had been viable. This includes algorithms for detecting and removing attacks and anonymization. This is considered a high-risk task. The solution would be to implement only a subset of possible analyzers and anonymizers. Second, the coding may turn out to be more advanced than first expected. This is of course a risk also associated with the choice of implementing a Bro extension. This is considered a low-risk task. To overcome this risk, learning necessary C++- and Java-techniques will start at an early stage. Java will be used in the case of a standalone prototype and C++ in the case of a Bro or NetDude extension. If the coding process would fail, a framework of the system with class frameworks, interfaces and function definitions will be presented. This will be described in relation to the methodology devised.

The IDS metric calculations part brings only low-risk tasks. The calculations itself will present no risks since established methods for calculations will be used. The task of providing data for calculations may though be a problem. The prototype is meant to provide a clean test data set. This data set will be tested against an IDS and the false positive rate will be calculated. To compare, a test data set based on generated traffic will also be tested by the same IDS. Probably will also the same recorded test data set as was used as input to the prototype be tested by the IDS, to compare results between the cleansed and not cleansed data sets. If the prototype can not be finished, no calculations providing results valuable to the thesis can be made. In this case the relation between false positive rates of the same test approaches mentioned would be assessed based on literature about the topic.

The greatest risk may be lack of time. A common experience is that things always last longer than expected. How long the different tasks last depend on the difficulty of
### Table 3: Risks

<table>
<thead>
<tr>
<th>No.</th>
<th>Task</th>
<th>Prob.</th>
<th>Imp.</th>
<th>Countermeasure</th>
<th>Decision date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Devising methodology</td>
<td>Low</td>
<td>Med</td>
<td>Reduce scope</td>
<td>24th Feb, 2006</td>
</tr>
<tr>
<td>2.1</td>
<td>Prototype based on Bro or NetDude API</td>
<td>Low</td>
<td>Med</td>
<td>1. Turn to standalone prototype 2. Describe prototype based on Bro or NetDude in words</td>
<td>17th Feb, 2006</td>
</tr>
<tr>
<td>2.2</td>
<td>Standalone prototype - complete system</td>
<td>High</td>
<td>Med</td>
<td>Implement prototype for a subset of protocols</td>
<td>6th Mar, 2006</td>
</tr>
<tr>
<td>2.3</td>
<td>Coding skills</td>
<td>Low</td>
<td>Med</td>
<td>1. Learn languages at an early stage 2. Present framework of system</td>
<td>6th Mar, 2006</td>
</tr>
<tr>
<td>4</td>
<td>Time</td>
<td>Med</td>
<td>Med</td>
<td>Limit scope, reduce prototype functionality</td>
<td>27th Mar, 2006</td>
</tr>
</tbody>
</table>

Each task and the types of problems encountered. If serious problems are encountered at each task this might lead to problems finishing the project as planned. At this time of writing each task is thoroughly thought through. Therefore this scenario is much unlikely. If time will be a problem the scope of the project will be reduced, especially in regard to the prototype. Instead of developing a functional prototype a skeleton prototype would be presented for further development.
8 Ethical and legal considerations

One of the goals of this project is to develop a prototype for network data anonymization. To test the prototype, network traffic must be recorded. This will be done in the Sandbox Laboratory on a line connected to the Uninet network. Permission for this is granted from the Information Technology Services Department at Gjøvik University College. Use and handling of data containing private or sensitive information is governed by The Norwegian Privacy Legislation [21]. No test data sets will be distributed. The test data sets will only be used for test purposes.

There are also ethical considerations in regard to developing a system for anonymizing network traffic. The anonymized traffic is meant to be distributable. If there exist uncertainty if private information remain in the cleansed data set, the data set should not be distributed. This is why a methodology and system for anonymization always should contain a human inspection module. The methodology devised in this thesis will contain such a module.
Bibliography


