

Guided Research Proposal

Anonymization of SNMP traces

Matúš Harvan
m.harvan@iu-bremen.de

Spring Semester 2005

1 Executive summary

Simple Network Management Protocol (SNMP) is a protocol to access management and control information of network devices. It is a very lightweight protocol capable of easily monitoring thousands of devices simultaneously. Therefore, SNMP is used extensively in enterprise networks and by ISPs, especially for monitoring purposes.

It is believed that SNMP is used differently in different environments and various SNMP agents perform differently. People guess where possible problems might be and do optimizations for assumed bottlenecks. However, it is not understood how exactly SNMP performs in practice, what are the various interactions and where exactly the real problems are as there is no data available to the research community from operators of large networks. The operators are concerned about the privacy of their networks' users and afraid of providing potential attackers with sensitive information about their network allowing for easier break-ins.

The proposed project aims at anonymization of SNMP traces so that SNMP traces could be made available. Being able to remove sensitive data out of SNMP traffic traces and anonymize these traces in such a way that privacy or security of the originating network would not be endangered while still leaving enough information in the anonymized traces to be useful for network research, would be of great help. Obtaining these traces would help clarify how exactly SNMP is used, allow to study interaction patterns of different SNMP implementations, compare performance and evaluate different SNMP approaches.

2 Summary description

Simple Network Management Protocol (SNMP) is a protocol to access management and control information of network devices. It is a lightweight, stateless protocol capable of easily monitoring thousands of devices simultaneously without significantly loading the involved hosts. Therefore, SNMP is used extensively in enterprise networks and by ISPs, especially for monitoring purposes. Furthermore, large amounts of new MIB modules are being produced (on customer requests) by companies as Enterasys, Juniper or Cisco[5], further showing the popularity and wide deployment of SNMP.

It is believed that SNMP is used differently in different environments and various SNMP agents and management systems perform differently. In particular, there are several possible ways to highly optimize SNMP interactions and data retrieval as outlined in [9]. However, due to the lack of available traces from operational Internet networks it is not known how exactly SNMP is used in practice, which particular management applications are preferred and how efficiently they make use of available protocol options.

Traffic trace owners hesitate to make their traces available as they are concerned about the confidentiality and privacy of the information contained in these traces, providing potential attackers with information about their networks such as IP addresses of specific servers, parts of network topology or the possibility of identifying from which network a trace comes. Being able to remove sensitive data out of SNMP traffic traces and anonymizing them in such a way that privacy or security of networks would not be breached, while still leaving enough information in the anonymized traces to be useful for SNMP analysis might significantly improve the availability of the traces. Obtaining these traces would allow to analyze the usage, behavior and interaction patterns of SNMP in real-world networks.

The proposed project aims at trying to anonymize SNMP traces in such a way that network operators would not have to be concerned about making their traces available for network research while the anonymization would still maintain enough information in the traces to keep them useful for analysis of SNMP.

3 Statement and motivation of research question

The aim of this project is to investigate on possibilities of anonymizing SNMP traces. This topic is of particular interest as no traces of SNMP traffic from real-world networks are available, rendering analysis of SNMP in real networks almost impossible. Therefore, it is not understood how exactly SNMP is used in practice and how protocol options are being used by management applications. This leads to optimizations done for assumed interactions while not being sure if and where exactly optimizations would be needed. The main reason for the absence of SNMP traces is the concern of network operators about disclosing sensitive information by providing the traces for networking research. However, having a suitable anonymization scheme would be likely to relieve the current reluctance to provide an insight into their networks and give researchers access to anonymized versions

of the traces.

There are two particular challenges involved in anonymizing SNMP traces. The first challenge is to find a lexicographic-order- and prefix-preserving IP address transformation. It has to be a prefix-preserving transformation so that the anonymized trace would still be usable if prefix relationships were important. The lexicographic-order-preserving requirement comes from the way how SNMP works. Larger objects like tables (potentially indexed by IP addresses) are stored in lexicographic order and are sequentially retrieved via several smaller queries. In order for these SNMP interactions to be recognizable in anonymized traces, the IP anonymization scheme has the additional requirement of being lexicographic-order-preserving. The prefix-preserving part is solved by tools like *tcpdpriv* [2] (using the `-A50` option, but unfortunately being susceptible to an attack described in [10]) or a more secure tool *Crypto-PAn* [8], implementing a cryptography-based scheme described in [7] and not suffering from *tcpdpriv*'s weaknesses. None of these approaches, however, treats the preserving of lexicographical ordering.

The second challenge is to anonymize the actual SNMP trace, especially the payload contained in the network packets. Tools as *tcpdpriv* and *Crypto-PAn* operate only on IP, TCP and UDP headers, but do not focus on the actual payload, which is application-protocol specific. The tools only scramble, hash or encrypt the whole payload rather than parsing it. However, for anonymized SNMP traces to be useful, the SNMP payload has to be anonymized in a way that keeps the structure intact and only anonymizes certain fields. Obviously, an IP transformation scheme, as described in the first challenge, is needed as one of the most important fields to be anonymized is IP address. Similar projects exist, successfully dealing with anonymization of FTP traces [3] or router configuration files [1], but according to my knowledge there is no project for anonymization of SNMP traces including anonymization of the SNMP payload.

In the proposed project, I would like to address the following questions:

1. Is it possible to create a lexicographic-order-preserving and prefix-preserving IP address transformation?
2. Is it possible to anonymize SNMP traces (including the payload) while still keeping enough features to allow for SNMP analysis?

4 Planned experiments/investigations

In order to address the first part of the research question (lexicographic-order- and prefix-preserving IP address transformation), I would like to investigate on the possibility of modifying the cryptography-based scheme explained in [7] to allow for lexicographic-order preserving. I would also like to see under which conditions (IP address space coverage) such a transformation may be possible and what would be the impact on security of using such a scheme.

To tackle the second part of the question I intend to extend the *snmpdump* tool [4] (providing parsing of pcap format SNMP traces and conversion to an XML-based output).

In order to anonymize the XML output, I would like to use *libsmi* [6] for parsing MIB descriptions and making *snmpdump* aware of the semantical meaning of traces – at the moment it is aware only of the syntactical one. This should allow *snmpdump* to automatically decide which parts of SNMP payload should be anonymized and in what way (eg. which parts might contain an IP address or other sensitive data). For the anonymization, I would like to use the “filter-in” principle used in [3], where only things known to be non-sensitive are allowed to pass the anonymization unchanged. This should guarantee easy verifiability of the anonymization process.

4.1 Time Plan

- Find a lexicographical-order-preserving and prefix-preserving IP address anonymization or show that it is not possible: 2005-03-26
- Finish implementation of anonymization tool: 2005-04-18
- Final report: 2005-04-25

5 Expected results and evaluation criteria

Ideally, I would like to solve both parts of the research question and come up with a tool capable of reliably anonymizing SNMP traces or define conditions for how well a trace can be anonymized. However, succeeding only in one part of the research question would also be helpful as further research could then build upon the initial findings.

The expected result of the project would be a lexicographic-order- and prefix-preserving scheme for IP address anonymization and a tool for anonymization of SNMP traces in a suitable way (removing all sensitive information but keeping enough characteristics to be useful). Should such an anonymization scheme not be possible, I would like to show (in a rigorous way) that it is not possible or under what circumstances it is possible. Should the project succeed and should we obtain a tool capable of reliably anonymizing SNMP traces, it might be possible to reach an agreement with operators of several larger networks to provide us with anonymized SNMP traces. This could then lead to projects similar to [3] or [1].

References

- [1] David A. Maltz, Jibin Zhan, Geoffrey Xie, and Hui Zhang. Structure preserving anonymization of router configuration data. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, 2004.
- [2] Greg Minshall. *tcpdpriv*, 1996. <http://ita.ee.lbl.gov/html/contrib/tcpdpriv.html>.

- [3] Rouming Pang and Vern Paxson. A high-level programming environment for packet trace anonymization and transformation. In *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, 2003.
- [4] Jürgen Schönwälder. snmpdump, 2004. <ftp://ftp.ibr.cs.tu-bs.de/pub/local/snmpdump-0.1.0.tar.gz>.
- [5] Jürgen Schönwälder. Characterization of SNMP MIB Modules. In *9th IFIP/IEEE International Symposium on Integrated Network Management*, Nice, May 2005.
- [6] Frank Strauss. libsmi, 1999. <http://www.ibr.cs.tu-bs.de/projects/libsmi/>.
- [7] Jun Xu, Jinliang Fan, and Mostafa H. Ammar. Prefix-preserving ip address anonymization: measurement-based security evaluation and a new cryptography-based scheme. In *Proceedings of the 10 th IEEE International Conference on Network Porotocols (ICNP'02)*, 2002.
- [8] Jun Xu, Jinliang Fan, Mostafa H. Ammar, and Sue Moon. Crypto-pan, 2003. <http://www.cc.gatech.edu/computing/Telecomm/cryptopan/>.
- [9] W. Yeong. SNMP Query Language. Technical Report 90-03-31-1, Performance Systems International, March 1990.
- [10] Tatu Ylonen. Thoughts on how to mount an attack on tcpdpriv's "-a50" option... <http://ita.ee.lbl.gov/html/contrib/attack50/attack50.html>.