Dipartimento di Informatica
Università del Piemonte Orientale “A. Avogadro”
Spalto Marengo 33, 15100 Alessandria
http://www.di.unipmn.it

Policy-Based Anonymous Channels

*Authors: Lavinia Egidi (lavinia.egidi@mfn.unipmn.it),
Giovanni Porcelli (giovanni.porcelli@mfn.unipmn.it)*

TECHNICAL REPORT TR-INF-2005-11-05-UNIPMN
(November 2005)

The University of Piemonte Orientale Department of Computer Science Research Technical Reports are available via WWW at URL http://www.di.mfn.unipmn.it/.

Plain-text abstracts organized by year are available in the directory

Recent Titles from the TR-INF-UNIPMN Technical Report Series

2005-04 *An Audio-Video Summarization Scheme Based on Audio and Video Analysis*, Furini, M., Ghini, V., October 2005.

2005-03 *Achieving Self-Healing in Autonomic Software Systems: a case-based reasoning approach*, Anglano, C., Montani, S., October 2005.

2005-02 *DBNet, a tool to convert Dynamic Fault Trees to Dynamic Bayesian Networks*, Montani, S., Portinale, L., Bobbio, A., Varesio, M., Codetta-Raiteri, D., August 2005.

2005-01 *Bayesan Networks in Reliability*, Langseth, H., Portinale, L., April 2005.

2004-08 *Modelling a Secure Agent with Team Automata*, Egidi, L., Petrocchi, M., July 2004.

2004-07 *Making CORBA fault-tolerant*, Codetta Raiteri D., April 2004.

2004-06 *Orthogonal operators for user-defined symbolic periodicities*, Egidi, L., Terenziani, P., April 2004.

2004-05 *RHENE: A Case Retrieval System for Hemodialysis Cases with Dynamically Monitored Parameters*, Montani, S., Portinale, L., Bellazzi, R., Leonardi, G., March 2004.

2004-04 *Dynamic Bayesian Networks for Modeling Advanced Fault Tree Features in Dependability Analysis*, Montani, S., Portinale, L., Bobbio, A., March 2004.

2004-03 *Two space saving tricks for linear time LCP computation*, Manzini, G., February 2004.

2004-01 *Grid Scheduling and Economic Models*, Canonico, M., January 2004.

2003-08 *Multi-modal Diagnosis Combining Case-Based and Model Based Reasoning: a Formal and Experimental Analisys*, Portinale, L., Torasso, P., Magro, D., December 2003.

2003-07 *Fault Tolerance in Grid Environment*, Canonico, M., December 2003.

2003-06 *Development of a Dynamic Fault Tree Solver based on Coloured Petri Nets and graphically interfaced with DrawNET*, Codetta Raiteri, D., October 2003.

2003-05 *Interactive Video Streaming Applications over IP Networks: An Adaptive Approach*, Furini, M., Roccetti, M., July 2003.

2003-04 *Audio-Text Synchronization inside mp3 file: A new approach and its implementation*, Furini, M., Alboresi, L., July 2003.

Policy-Based Anonymous Channels

Lavinia Egidi Giovanni Porcelli

Dipartimento di Informatica
Università del Piemonte Orientale *A. Avogadro*
Spalto Marengo, 33 - 15100 Alessandria - Italy
{lavinia.egidi, giovanni.porcelli}@mfn.unipmn.it

Abstract

In view of the legislations on privacy, we address the issue of limiting non-technical responsibility of the administrators of a computer network. A primary concern regards the anonymity of channels within a local network. Solutions proposed in the literature increase significantly the traffic. In this paper we argue that it is possible to implement anonymous and private channels in a local network, without increasing the traffic, with a suitable combination of an appropriate policy and technical tools.

1 Introduction

Article 8 comma 1 of the Charter of Fundamental Rights of the European Union [7], and now also Art. 50 of the Draft Treaty establishing a Constitution for Europe [11] state that everyone has the right to the protection of personal data, and that a European law shall lay down the rules relating to the protection of individuals with regard to the processing of personal data. Council Resolution [12] stresses among other things the relevance of security issues. The laws of Member States of the European Union must comply with EC directives For instance, the Italian law on privacy, L. 196/03, [10], is based on the same fundamental principles, and besides addressing privacy concerns, it defines the minimal security measures that must be adopted in the management of sensitive data.

Therefore, technical staff administering a network is on the one side compelled to offer service of a good quality and at the same time has the duty to respect the privacy of users. They find themselves burdened with the knowledge of sensitive data relative to the users, whereas this is neither strictly nor necessarily an issue they are qualified to deal with. We address privacy issues specifically focusing on the concern of protecting the rights of technical staff. We believe that the sensitive information on the customers, that is in any case known to some critical section of the service provider, must be made known elsewhere only as strictly necessary.

We addressed the issue from the point of view of application servers in [13] and [14]. In those papers, we showed under which hypotheses it is possible to implement the essential services in such a way that an attacker with knowledge of all configuration information that is available on the system needs a prohibitive amount of resources in order to infer sensitive information. The consequence is that the problem of anonymity is moved to the communication channels.

In this paper we study the possibility of implementing anonymous channels within a local network. (We assume a TCP/IP network over a fully switched Ethernet.) We discuss to what extent anonymous channels can be satisfactorily implemented defining a suitable policy.

We give a definition of privacy in terms of the administrator's view of the system. This approach is justified by our interest in the rights of technical staff. But also notice that in a fully switched Ethernet, normally other users do not represent a threat to the privacy of communication channels.

We point out and discuss that the policy on collection and usage of log files plays a central role in shaping the administrator's view of the system. The underlying idea is to minimize the usage (or availability) of log files, still maintaining the critical ones that are necessary for network management.

Thus, we analyze virtual channels at different levels of the TCP/IP stack, questioning that the corresponding log files should be indispensable. As a consequence, we propose a scenario of network administration in which the monitoring of traffic and the legal treatment of data is restricted in such a way that the information collected is sufficient for a functional management of the local network, doesn't disclose sensitive data and allows for emergency security interventions.

There are various proposals in the literature, for the implementation of anonymous channels. They essentially boil down to families of proxies (mixers [8, 22, 16, 17, 19]) and the use of broadcast along with encryption (see e.g. [1]). Both systems tend to increase the traffic in the system and the load on the machines.

The simpler proxy solution must be supported by a suitable policy on the logs. For instance, the system JAP [19], for anonymity and privacy over the Internet, requires that mix providers sign an official declaration, that they do not save connection log files or exchange with other mix providers data which could be used to uncover JAP users. This must be equivalently enforced in a local network.

An alternative uses covert channels, tunnelling other communication in http-traffic[3]. This solves traffic analysis issues, but does not respond to the privacy goals we set.

Instead of mixers, one might think of using agents: autonomous travelling agents would carry messages, and they could hide source and destination of the data they carry by serving many masters at once, and taking random paths to reach their targets. This solution might also lead to traffic increase and would require techniques for securing agents (the latter are still the object of much research; existing solutions are preliminary and expensive [23, 6, 2]).

In contrast to these approaches, we suggest that the definition of a precise policy, backed by the careful reorganization of the network administration, and suitable technical tools, would solve the problem imposing a minimal overhead on the network.

We believe that a suitable policy can be reasonably enforced, using appropriate technical tools. However, it must be kept in mind that the cost of anonymity is, in our proposal, paid in terms of making the network management policy more restrictive. Whether this is desirable or not is a trade off that involves many parameters, and a political decision.

Security issues are out of the scope of this paper. It should be understood that policy definition and adoption depend on the specific services that are offered and on the level of security that is required. This also conditions the possibility of offering anonymity at all—in the administrative LAN of a bank, user anonymity might be out of the question.

The remainder of this paper is organized as follows: in the next section we link the issue of anonymous channels with the view of the system of network administrators, and bring up the related issue of logging policies. In Section 3 we define our basic assumptions on the network. In Section 4 we argue that anonymity of the communication channels can be achieved in the local network. We give a more concrete picture of our proposal in Section 5, in which we analyze a possible scenario of network management. We conclude with some remarks on future work.

2 Limited and instantaneous view

We ground our analysis on the following observations. First of all, absolute privacy requires protocols that are very different from what we are normally used to, and thus undesirable; instead, privacy should be obtained as a combination of protocols and policies [20]. But, if privacy of data can be infringed with sporadic and instantaneous actions, it is hard to devise a monitoring strategy that exposes violations of the policy. On the other hand, whenever retrieval of sensitive data requires a continuous and/or ubiquitous collection of information from the network, with suitable technical tools the policy can be enforced.

Therefore, our aim is to prove that the view of the system of technical staff can be reduced to a local and/or istantaneous view in all domains that involve sensitive information, without impairing network management.

For proper network management, the system administrator must be able to monitor traffic in real time in case of need. The permission must be granted via an *efficient* procedure, in order to avoid costs that would be caused by delays. The use of virtual private channels at IP level within the LAN (implemented with protocols like, e.g., IPsec) would not change the view of the system administrator, who would have the means to access the traffic in the clear. As a benefit, it would protect from third parties snooping on the LAN. But, since, as mentioned, security issues are out of the scope of this paper, we confine ourselves to the observation that the effect of such channels on the administration would be only to make it heavier.

Notice that real-time monitoring doesn't endow the system administrator of an ubiquitous and continuous view of the system. Log files can. But security concerns impose the collection of log files.

2.1 Log files and the policy

When indiscriminate use of the log-files is admitted, log files represent the continuous and ubiquitous view of the system that we want to avoid. Therefore it is necessary that data be logged and managed with different policies, depending on how frequently they must be accessed and on the privacy of the data they contain. Policies must specify which parts (which headers and/or payloads) of each packet can be stored in log files with different (privacy based) usage profiles.

In detail, the following steps must be taken:

- define clearly what information must be normally available to technical staff;
- provide technical tools that separate in a clear way sensitive data from technically valuable data, so that data necessary for the correct functioning of the system are always available to technical staff, and other data can be collected in an emergency situation;
- define clearly which operations are legal for technical staff in conditions of normal operation;
- provide means to monitor the activity of technical staff in order to establish whether they are trying to access data illegally; this, for their protection and with forensic aims, in order to be able to clear a honest administrator in case of doubt, and the opposite;
- provide efficient emergency procedures for dealing with unusual and critical situations.

A good policy, as always, is a policy that can be easily enforced; therefore it must be defined with a view to the techniques available for monitoring usual activity and detecting misuse. The limits that must be posed depend strongly on the tools that are to be used to monitor the activity of technical staff. These tools, in turn, depend strongly on issues of secure operating systems. (This is an area of independent interest and we plan to investigate it further as future work.)

The different policies can be implemented taking advantage of syslog flexibility, including features like priority tags. Syslog classifies messages according to their criticality, labelling them with tags. The first few tags in descending relevance order (EMERG, ALERT, CRIT) label events that must definitely receive attention. Also some ERR messages should be attended to. We maintain that logs of conditions of such priorities should always be available to the system administrator. Messages tagged with lower priority (WARNING down to DEBUG) can be forgotten in everyday administration. On the other hand, it is necessary to define procedures that allow the collection and usage of logs with messages of any priority level (for instance, do not ask an administrator to install sendmail without looking at DEBUG level messages!). Should it be necessary to collect more critical logs, even during normal operation, classified logs should be encrypted or redirected to a protected machine via, say, a serial connection.

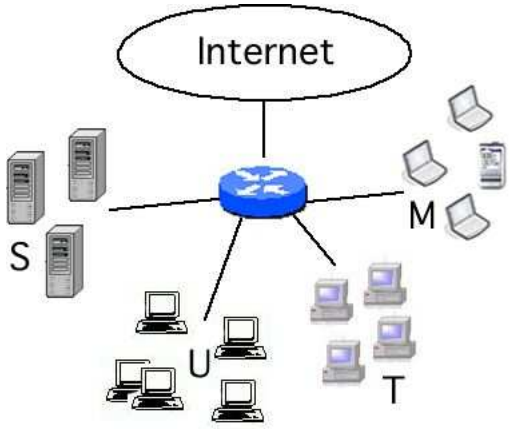


Figure 1: The network is organized in subnets (S = servers; U = hosts administered by users; T = hosts administered by technical staff; M = mobile devices).

Many tools now enable a system administrator to centralize all relevant logs to carry on suitable analysis on them. Therefore the policy must also define which kind of cross analysis of logs is legal. Policy implementation here includes providing properly configured tools for cross analysis.

3 Configuration

Client hosts.

Our discussion assumes that client machines are "private". This is no light assumption and requires some discussion per se. When a new desktop is connected to the network, two are the possible policies that can be followed. Either the user administers the machine him/herself, or administrator privileges are granted to the technical staff that is in charge of the machine. In the first case, liability of the system administrator is not extended to the private machines

The second choice impairs privacy, altough it can be argued that with a role based administration of the machine (we think, e.g., of Trusted Solaris [25]), and enforcing a suitable policy, it should be possible to preserve user privacy to some extent.

The network.

We assume a TCP/IP network, over a fully switched Ethernet. Moreover, the local network is organized in subnets (Fig. 1). This enables us to customize policies to subnets, depending on different security and privacy requirements.

We assume that servers are on one (or more) separate subnet(s) (Fig. 1, S). User-administered and staff-administered client hosts are connected to two separate subnets (Fig. 1, U and T), because in the first case a lower reliability level can be guaranteed. (In our LAN it has happened that users administering their own computer chose to run their own DHCP server—when one was already active in the network—or misconfigured the IP number of their laptop, setting it to the gateway's IP.) A more stable network access is paid in terms of privacy in the subnet with staff-administered machines.

Even more problems arise in a subnet to which mobile devices are allowed to connect (Fig. 1, M). Reliability and security here are even lower, due on the one side to the fact that mobile devices are typically user-administered, and on the other on the higher exposure of mobile devices to various infections. Therefore such a subnet should be treated as an external network, for the purposes of security and trust.

4 Anonymous channels

We say that a communication channel is *anonymous* if it does not expose the relations between users and their activities [18].

This implies that the chain associating users to IP numbers to MAC addresses to applicative payload of packets must be interrupted in more than one point. It also implies that client/server relations in a connection should be hidden. Whereas all channels above transport level can be easily implemented as private and anonymous, with standard cryptographic techniques, it is not clear what can be done below that level.

Depending on which information (what headers) are collected in a log file, this can be regarded as providing continuous view at a some level of the TCP/IP stack (the correspondance with stack layers not being necessarily accurate). Therefore the following discussion includes log files.

4.1 IP level

The relation between IP numbers and users can be hidden using dynamic association of IP numbers, with a DHCP protocol.

It must be noted, though, that on many new firewalls, after an initial authentication phase, rights granted to users are associated to the user's IP. This clearly works in a direction opposite to our privacy goals, but is an aspect of the obvious trade-off between privacy and privileges.

On the other hand, whenever firewalls grant privileges based solely on the service requested, privacy is protected, even if the packet filtering mechanism is actually looking at IP source and destination of each packet, as well as at the ports. The latter (typical) situation does not jeopardize privacy as long as log files are not maintained, or (more realistically) log files with IP information are classified.

4.2 Physical layer

The lower one goes down the TCP/IP stack, the harder it becomes to hide information. The question is how much information is necessary for normal network management.

With an oscilloscope, for instance, one can analyze the signal travelling on the wires, and even identify a specific network card, by the characteristics of its transmission. We maintain that collection of data at this level, is not normally necessary. An occasional use of the instrument might be required by some functioning anomaly, but it would not sensibly impair privacy. A continued data collection with it should on the other hand be detectable. A suitable policy and monitoring should be sufficient to rule out such a threat to privacy at physical level.

4.3 Datalink layer

The association between the MAC address of a client host and the physical port to which the host is plugged in turns out to be much more critical than one would expect. In a network with high mobility this might not be an issue, but often users have their own desktop computers or plug-in their laptops in their office, and don't move around. In such a static environment, the MAC addresses are as good as the names of the users, unless the MAC address can be decoupled from the physical port.

For some machines, also the MAC address can be changed dynamically. This should not be done too often, or else the switches end up for behaving like hubs: before a switch has learned the port-MAC address association for a machine, it simply broadcasts all packets for that MAC address. Changing MAC addresses once or twice a day should be sufficient. Clashes of forged MAC addresses are unlikely, but could be prevented altogether with a centralized monitoring (centralized to each subnet), that allows connection to the subnet only after having checked that the MAC address that the new machine is proposing is not already in use in that subnet.

An interrogation of the switches allows to bind the MAC address to the physical port to which the machine is connected. This, as discussed, in environments with low mobility, allows to link a MAC address to a user. But proper configuration of the switches allows to restrict the IP from which such requests are allowed and to log all connections: this way an effort towards building and keeping up-to-date associations between users and the MAC addresses of their computers can be exposed.

In a more standard setting in which MAC addresses are fixed, the association physical port-MAC address is essentially always known, and can therefore be regarded as public information. Now, if policies specify that (unclassified) datalink level logs must only preserve datalink headers (no other headers and no payload), then privacy is again safeguarded by suitable enforcement and monitoring. Indeed, recall that we have placed servers on a separate subnet. Therefore datalink layer headers of packets of a client/server connection never contain both the client and the server MAC addresses, one of source or destination always being the router.

Of course, cross examinations of logs, would enable to derive associations as to which machine has accessed which

User name	IP address	MAC address	port
Arthur Dent	193.206.52.42	4f09ac9ce211	3,A,74
Ford Prefect	193.206.52.24	9ad2ff1a423b	1,D,127
. . .	. . .	. . .	. . .

Figure 2: User IDs through the TCP/IP stack.

server, even working at datalink level.

5 Minimizing logs and static information

In this section we analyze some specific privacy threats in a possible scenario of network administration inspired by the situation in our University and with reference to specific tasks. We propose policy based, privacy aware alternatives.

5.1 User–IP–MAC association

Consider Table 2. It summarizes for all users in a network the IP address of their machine, the corresponding MAC address and the physical port to which the machine is plugged. The port here is expressed as a sequence describing the path to the physical port through the tree of switches (the organization of the tree matches the topology of the building; thus the sequence indicates a floor number, a section of the building and the number of the port in that section).

The table enables essentially to associate user identities as they appear at different layers of the TCP/IP stack. It is extremely useful for the following purposes:

1. **Authentication:** beforehand knowledge of MAC addresses provides a simple authentication tool. When a new machine tries to connect to the network, its MAC address is checked: if it is listed in the table, the connection is accepted. This technique can be used in a rigid fashion: the table is static and therefore the port to which each host may connect is fixed in advance. Alternatively, one can accept that the port entry be filled after the connection, say via an appropriate request to the switches.

2. **Tracing troublemakers:** it allows to trace the port and MAC address corresponding to an IP number which is causing some problems on the network (a faulty or a malicious machine). With the table at hand, even in a large environment, it is the matter of seconds tracing the offending machine.

3. **Troubleshooting:** it can be useful to solve problems for a user who is experiencing some difficulties with services. In this case, the table is very practical to retrieve information on the topology of the network.

The table provides most of the links in the chain associating users with their activities, and therefore is at odds with the pursuit of anonymous channels. This leads to the following considerations:

- **Table destruction:** We do not want to keep such information, and we must also ensure that it is possible to monitor technical staff to ensure that they don't collect such information.
- **Information retrieval:** On the other hand, it must be possible to reconstruct the necessary portions of the table each time that some technical problem arises and must be solved.
- **Dynamical associations:** Finally, we must avoid that the collection of sensitive data on one occasion impairs the privacy of a user at all subsequent times (i.e. the sensitive relations must change dynamically sufficiently often).

Destroying the table amounts to breaking the associations between columns, or at least rendering them useless. As we discussed in the previous section, the IP address can be decoupled from all other information in the table, using a dynamic association with a DHCP server. Where forged MAC addresses are possible, their association with users can be eliminated as well. Also recall that, even though in general we cannot assume the latter, locating servers on a separate subnet allows to hide at datalink level sensitive information on the activities of users.

Information retrieval must be finalized to completing the Tasks 1 through 3 that motivate the maintainance of the table in the first place.

Authentication can be solved with the adoption of more sofisticated authentication techniques. Solutions as (e.g.) the system described in [5], Authipf [4] and Cisco Secure User Registration Tool [9], not only allow a privacy aware management of the network, but can also provide a stronger authentication service. Basically, the idea is that a machine must authenticate upon connection to the network. The authentication data is *not* the host's MAC address (which, as observed, is even forgeable), but some other secret that the parties have agreed upon. This allows a more flexible authentication procedure, allows to connect machines to different (virtual) LANs, and to any port (maybe with restrictions that depend on virtual LANs). Logging of authentication sessions, as well as MAC–IP associations should be logged in classified files.

Tracing the physical port of a computer that is causing trouble on the network from its MAC address can be solved by telnet connections to the switch(es). Notice that, with proper configuration of the switch, such connections can be limited to specific IPs and can be logged. This way a continuous activity of collection of information (basically attempts at keeping an updated version of the table above at all times) can be exposed.

Tracing a MAC address from an IP address is more tricky, especially if we want to preserve privacy and minimize the availability of log files. Indeed, if the faulty behaviour is noticed in a subnet different from the one to which the troublesome host belongs, the host's MAC address is not available on the packets. The policy can specify that system administrators may temporarily block traffic from a faulty or malicious host (unless the host is a server, of course). This is acceptable, since the first goal is the correct functioning of the network. It is very likely that, after the traffic has been blocked, the user that works on that host will pretty soon report to the system administrator.

Relevant information on the topology of the network for **troubleshooting** purposes is solved snooping packets and requesting information of switches, after obtaining user consent to all these operations. Since IP addresses are assigned dynamically, the exposure of the IP that the user is currently using has only temporary significance. The same holds for the exposure of the MAC address, in case this is assigned dynamically (as discussed, wenever MAC addresses are statical, they can be regarded as public information).

5.2 Automatic processing of log files

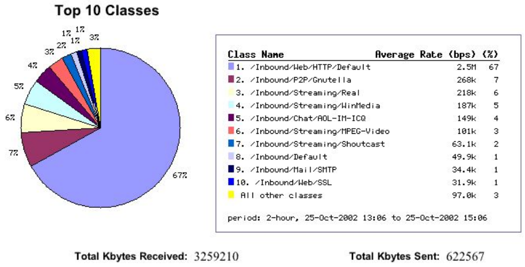
Log files are very important for the solution of technical problems that may occur on the network. But in most cases their value lies on statistical or general information derived from them, rather than on the specific data relating to a single connection or a single user. This implies that most of the time summaries highlighting some features of the traffic are sufficient to the system administrator, and the sensitive details pertaining to single users can be kept in store as classified data, to be available only in emergency situations, subject to specific authorizations. These summaries can be produced by automatic tools, thus eliminating the need of human inspection of sensitive data.

We refer in the following to some specific instances.

Network and transport layer logs.

Log files in which data from the network and transport layer are present can convey sensitive information even though the IP numbers cannot be related directly to users. The reason is that this data could be in principle useful for user profiling. Most critical is here perhaps the converse: the known profile of a user can help to identify traffic as pertaining to that user and therefore to expose details of the user's activity. In a large network in which IP numbers are associated dynamically, such a (direct or reverse) profiling activity is not straightforward. Information is in any case leaked, which we want to avoid as much as possible. So the question is how relevant logs that contain network and transport headers are for system administration.

One first fact is that these logs are useful, for instance, for deriving information on the kind of traffic that flows through the local network, classified according to the different services. This information is essential in case of network congestion. Automatic traffic-shaping tools take care of the congestion problem in a flexible way, based on priorities that are defined for various services and subnets. But such tools can also be used to produce all sorts of statistics, depending on the configuration. Processed files can contain completely anonymous data (see e.g. Fig. 3) that not only respect the users' privacy but are also largely more useful for network administration. Of course they can also be configured to trace traffic pertaining to a single user, but this will be forbidden by the policy and the configuration of the traffic-shaping tools can be periodically monitored.



Class Name	Average Rate (bps)	(%)
1. /Inbound/Web/HTTP/Default	2.5M	67
2. /Inbound/P2P/Gnutella	268k	7
3. /Inbound/Streaming/Real	218k	6
4. /Inbound/Streaming/WinMedia	187k	5
5. /Inbound/Chat/AOL-IM-ICQ	149k	4
6. /Inbound/Streaming/MPEG-Video	101k	3
7. /Inbound/Streaming/Shoutcast	63.1k	2
8. /Inbound/Default	49.9k	1
9. /Inbound/Mail/SMTP	34.4k	1
10. /Inbound/Web/SSL	31.9k	1
All other classes	97.0k	3

period: 2-hour, 25-Oct-2002 13:06 to 25-Oct-2002 15:06

Total Kbytes Received: 3259210 **Total Kbytes Sent:** 622567

Figure 3: Anonymous statistics on sensitive traffic. (The figure is from [21].)

DNS logs.

DNS activity is even more critical in connection to user profiling. For instance each user visits more frequently a number of web sites more relevant for his/her activity, and web accesses produce DNS queries. But the usefulness of complete DNS logs is questionable during normal operation. One needs to monitor errors and attemps at unauthorized zone transfers, but certainly not single user requests. Again, statistics might be useful for better knowledge of the traffic through the network, but those data can be utterly anonymous and produced by automatic processing tools.

IDS.

Intrusion detection systems (IDS) can also be employed for network diagnostic. To give a concrete example we refer again to incidents that occurred in our network when a user decided to run his own DHCP server, when one was already in use in the LAN; or when a user configured her laptop to use as IP address the gateway IP address. Our way to detect similar (catastrophic!) anomalies is to run the Snort IDS [24], configured to signal the presence of more than one DHCP server, or gateway. In general, an IDS can be configured to produce statistics and summaries of activity of the network, not necessarily oriented towards intrusion detection.

Cross analysis.

The association of information like CPU load, memory load, interface status, single deamons' errors, all linked to each other, is precious for understanding what goes on in the network. This data would help to expose many faults and errors in the network, without affecting privacy issues. Such data normally results from cross-examining multiple log-files. We argue that if the processing is done automatically by a properly configured tool, privacy is preserved.

Cross analysis of log files is otherwise, in our opinion, seldom necessary. In our experience, it might be useful in two or three instances in one year, in a network with roughly one thousand hosts.

For instance, we needed it to solve a puzzle when we realized that, configuring machines in a subnet with addresses belonging to another subnet, the network was still correctly working. The anomaly turned out to be caused by a configuration of the router, that set it up to work at datalink level when possible, for performance. As a result, the router didn't even look at the IP addresses of packets when they were directed to subnets that it was directly connected to.

We believe that it is acceptable to resort to an emergency procedure in such rare special cases.

5.3 Log files on servers

This paper focuses on the channels. But channels end at servers, and are used to convey services; therefore we cannot avoid a discussion of what is exposed by logs of processes.

Log files normally do not store the packet payload. At least this is the default choice of tools like TCPdump. Besides, when anonymity and privacy are the goals, it can be assumed that application data are encrypted. Therefore we will suppose that application data is never stored in log files.

Since we place the servers in a separate subnet, the MAC addresses of client hosts do not reach the server: all packets arrive with the router's MAC address. The IP addresses of users change dynamically, therefore are no issue either.

A real concern is more specific to the service that is being granted. If the service that is offered can be anonymous, then log files at the server cannot serve to link users with their activities. Notice that information can still be deduced by user profiling means; yet if the service is anonymous, the user ID changes at each access, as well as the IP address from which the connection is requested, therefore the profiling is not straighforward, unless the usage patterns are extremely distinictive.

For services that can be at most pseudonymous, like the use of an account or of e-mail service, we see no way of preventing knowledge of the activity of a single user. In [14] we explored the possibility of an anonymous e-mail service, but concluded that the latter can never substitute traditional e-mail, due to its several drawbacks.

Under the assumption that cross-analysis of log files is forbidden by the policy, log files at the server expose some information on the user activity only when the service itself is not anonymous. In the latter case, anonymity of the channel itself is obviously lost, but this would happen with anonymous channels obtained with any technology (even by magic!).

6 Conclusions

In this paper we discuss the possibility of defining anonymous channels in a local network through the adoption of a suitable policy and the monitoring of the administrators' activity. We argue that this approach is more realistic than other proposals found in the literature, because it doesn't increase the load on the network.

The persistency of the association of the MAC address to the physical port (and thus to the user) is obviously an undesirable weak point in our treatment. If it can't be avoided, we maintain that it can still be made harmless by the suitable definition of a policy.

Security issues were not addressed explicitly in this paper. For instance, we assumed that other users represent no threat to one's privacy in a fully switched Ethernet. This is of course false, if we take into account the possibility of attacks (see, e.g., [15]). But it is also true that using protocols like IPsec within the LAN, such problems can be solved. In short, although we do not discuss the interplay of privacy and security, we carried out our analysis in constant awareness of security issues. As a result, the solutions we propose here do not conflict with security based solutions, at least no more than privacy conflicts with security at all. We plan to draw the complete picture in the future.

We hope to foster with this paper a discussion on our alternative solution.

The logical next step of our research will be an indepth analysis of the flexibility and usefulness of tools like syslog and role-based administration, in view of our goals. We also must address performance concerns if we choose to encrypt sensitive logs. The final goal is to actually propose a policy respecting the constraints we set, for a specific network.

Finally, the case of wireless networks would require a specific treatment.

References

[1] M. Abadi. Private authentication. In *Privacy Enhancing Technologies*, pages 27–40, 2002.

[2] Joy Algesheimer, Christian Cachin, Jan Camenisch, and Günter Karjoth. Cryptographic security for mobile code. Technical Report RZ 3302 (# 93348), 2000.

[3] M. Bauer. New covert channels in http. In *CoRR*, volume cs.CR/0404054, 2004.

[4] R. Beck. Dealing with public ethernet jacks–switches, gateway and authentication. In *Procs. of LISA '99*, pages 149–154, 1999. `http://www.usenix.org/events/lisa99/full_papers/beck/`.

[5] A. Bussi. *Sistema Flessibile di Autenticazione di Computer per l'Accesso ad una Rete Locale*. (in Italian), Master's Thesis, Università del Piemonte Orientale, Italy, 2002.

[6] C. Cachin, J. Camenisch, J. Kilian, and J. Müller. One-round secure computation and secure autonomous mobile agents. In *Proc. ICALP'00*, LNCS 1853, pages 512–523. Springer-Verlag, 2000.

[7] Charter of fundamental rights of the european union. *Official Journal*, C 364:0001– 0022, 18/12/2000.

[8] D. Chaum. Untraceable electronic mail, return address, and digital pseudonyms. *Communications of the ACM*, 24(2):84–88, 1981.

[9] Inc. Cisco Systems. *Cisco Secure User Registration Tool 2.0*. 2001. `http://www.cisco.com/warp/public/cc/pd/wr2k/urto/prodlit/cregt_ds.htm`.

[10] Codice in materia di protezione dei dati personali, dl 196, 30/6/2003. *Gazzetta Ufficiale*, 174, s.o. 123, 29/7/2003. `http://www.camera.it/parlam/leggi/deleghe/testi/03196dl.htm`.

[11] The European Convention. Draft treaty establishing a constitution for europe. *CONV*, 850/03, 18/7/2003.

[12] European Community Council. Resolution on a european approach towards a culture of network and information security. *Official Journal*, C 048:0001 – 0002, 28/02/2003.

[13] L. Egidi and G. Porcelli. Minimal information disclosure in a centralized authorization system. In *First Workshop on Security Issues in Coordination Models, Languages, and Systems, SecCo03*, 2003.

[14] L. Egidi and G. Porcelli. Anonymity and certification: e-mail, a case study. In *Proc. of ACM SAC'04*, pages 397–403, 2004.

[15] Ettercap. `http://ettercap.sourceforge.net/`.

[16] P. Golle. Reputable mix networks. In *Proc. of the 2004 Workshop on Privacy Enhancing Technologies*, 2004.

[17] P. Golle and M. Jakobsson. Reusable anonymous return channels. In P. Samarati and P. Syverson, editors, *Proc. of the 2003 ACM Workshop on Privacy in the Electronic Society*, pages 94–100. ACM Press, 2003.

[18] D. Hughes and V. Shmatikov. Information hiding, anonymity and privacy: a modular approach. *J. of Computer Security*, 12(1), 2004.

[19] Jap anonymity and privacy. `http://anon.inf.tu-dresden.de/index_en.html`.

[20] M. Langheinrich. A privacy awareness system for ubiquitous computing environments. In G. Borriello and L.E. Holmquist, editors, *4th International Conference on Ubiquitous Computing, (UbiComp2002)*, volume 2498 of *LNCS*, pages 237–245. Springer-Verlag, 2002.

[21] Office of Information Technology. Illinois Wesleyan University, `http://titan.iwu.edu`.

[22] W Ogata, K Kurosawa, K Sako, and K Takatani. Fault tolerant anonymous channel. In *Information and Communications Security — First International Conference*, volume 1334, pages 440–444, Beijing, China, 11–14 1997. Springer-Verlag.

[23] T. Sander and C.F. Tschudin. Protecting mobile agents against malicious hosts. In *Proc. Mobile Agents and Security*, LNCS 1419, pages 44–60. Springer-Verlag, 1998.

[24] Snort, the open source network intrusion detection system. `http://www.snort.org`.

[25] Trusted solaris. `http://wwws.sun.com/software/solaris/trustedsolaris`.