PR3 Email Honeypot

Jean-Marc Seigneur, Anselm Lambert,
Patroklos G. Argyroudis, Christian D. Jensen

Jean-Marc.Seigneur@cs.tcd.ie, lambera@cs.tcd.ie,
argp@cs.tcd.ie, Christian.Jensen@cs.tcd.ie
Department of Computer Science
Trinity College, Dublin 2, Ireland.

Abstract: Although there are different tools and technologies available to prevent attacks on privacy when online applications are used, few tools are available for detection of actions that violate privacy agreements. The loss of privacy when third parties obtain email addresses of users without their consent can be followed by unsolicited emails – known as spam – sent on the open communication channel. In this case, the loss of privacy is aggravated by the cost spent by people having to go through all these email messages. However, spam can also be used to detect Websites that default on their privacy policy by giving away private information, such as email addresses. If spam is received by an email address that has only been given to a single Website, we may conclude that this Website has either given the address to the spammer or very poor security. In either case, registration with that Website should be avoided. In this paper, we present a tool to detect such exchange of subscribers' private email addresses between online providers and to check whether these providers have respected their advertised privacy policies.

Keywords: Email, privacy, honeypot.

1. Relations between Email and Privacy

Privacy is linked to intellectual and philosophical ideas. Privacy has many definitions in many domains, e.g. sociology and law. So it is difficult to define it precisely, encompassing all of its aspects. Cooley defined it as follows: the "right to enjoy life and be left alone" [10]. This definition implies that privacy may be threatened by some means. In fact, privacy information is valuable and different entities threaten this right to be left alone. For example, information can be used to build accurate user profiles for marketing and selling purposes but also to contact the person with the email address. New privacy vulnerabilities came along with the creation of the Internet, the Web and the electronic mail system. To some extent, personal information has become a commodity that can be traded in online commerce. The most sensitive personal information, called Personally Identifiable Information (PII), is directly associated with the real-world end-user identity. We see the email as a PII having the property to easily (i.e. for a very low cost) and effectively (i.e. an email delivered in the inbox will surely obtain human attention, even if it is only a fraction of time) make contact with the related human. Many different methods can be used to collect personal information from the online world: use of low level protocols (e.g. routes of IP packets or time of connections); high level protocol information (the Web browser's chattering [13], invisible hyperlinks [13], cookies); privileged positions of different actors able to monitor connections (telecom providers, Internet service providers, few major online user profiling companies compared to the number of interactions); higher level applications (email marketing [13], newsgroups [13], Web browser plug-in leaking information [23], other spyware). Regarding email, the loss of privacy when a third party obtains the email addresses of some people without their consent is usually followed by unsolicited messages – known as spam – sent on the open communication channel associated with the email address, which is the standard electronic mail system. This cost varies a great deal between users, but the overall cost is known to be very important. Actually, it is difficult to know how the spammer obtained the email because emails are shared between all entities that are, explicitly or not, allowed to use these emails. There are different means for spammers to obtain email addresses: if the email is stored on public Web pages or it has been used in newsgroups, spammers can harvest emails by searching them in those pages. In this case, no provider is really liable: users are almost responsible for exposure of their emails to spammers. However, if spammers have somehow obtained these emails from providers, those providers can be considered liable if their customers had opted-in for privacy policies stipulating that providers would not giving away private information. Of course, a provider can sell emails to spammers or "partners" with the provider's consent or those emails may be stolen from the provider's database without the provider's consent.

In general to protect privacy, the first line of defence has been based on data protection legislation: going from country-specific legislations to higher level legislations (e.g. European Union (EU) 95/46/CE [15]). However, in July 2002 the European Union nearly reversed its position regarding privacy by issuing a new directive on Privacy and Electronic Communications [16] that leaves each EU Member State free to adopt laws authorizing data retention. This was mainly due to the tragic terrorist attacks of September 11th 2001 in the United States and the fear of other terrorist attacks around the world. Even if some experts warned not to compromise privacy without having long term perspectives – "a key element of the fight against terrorism involves ensuring that we preserve the fundamental values which are the basis of our democratic societies and they very values that those advocating the use of violence seek to destroy" [12]. Thus legislation has shown its limits.

Another obvious tool for privacy is technology even though it strongly interacts with legislation as well as markets and social norms [21]. In this case, the trend is to give back more control to the owners of their privacy and personal information flows. Brunk defines "online privacy as having the ability to control information leaving you while online, and being able to exercise that control consistent with your values" [5]. He is actually researching features in software applications and Internet-based services that allow them to protect privacy with a focus on the human-computer interface. A specific example where more control may be given back to users is for digital identities and profiles information [7]. The notion of self-profiling is also relevant [27].

From a technological point of view, the current solutions for privacy protection exhibit five role categories [5]:

1. Awareness: features helping the user to understand privacy.
2. Detection: features looking for potential privacy problems often running in the background.
3. Prevention: features used as a precaution which are usually run when needed.
4. Response: features applying a countermeasure when a problem has been detected.
5. Recovery: features helping to get back to a "normal" state.

Prevention is the most common feature among the different solutions [5]. Another categorization is to use three themes: prevention, detection and avoidance [20]. Detection is still hard to achieve and detection tools are lacking [5, 20].

Currently, there is active research on privacy enhancing technologies (PET). New protections have been found at different levels: at the communication level [4, 9, 28]; at the system level [2]; at the application level [8]. The aim of PET is to minimize the collection of PII and to eliminate collection when it is not vital. There is currently a strong focus on policies, fairly exchanged or negotiated by both parties. Indeed, the Platform for Privacy Preferences (P3P) [11] defines a protocol where Internet users reveal their privacy preferences before they are allowed to access the information on the Internet if the privacy practices of the Web pages provider conform to these user's privacy preferences. One may argue that privacy-aware users would think twice before divulging PII, and in doing so minimize the collection of PII. Regarding email, it is now common for providers to describe their privacy policy in a Web page. During the registration process, the users are asked to read the privacy policy and to opt-in if they agree with the policy.

An important aspect of privacy is that people have dynamic privacy expectations: "our privacy needs change almost constantly in response to our desire to interact with one another and social moral and institutions affect privacy expectations" [5]. The change of the EU position regarding privacy protection, explained above, is one example of this dynamic aspect. Privacy is a trade-off "with efficiency, convenience, safety, accountability, business, marketing, and usability" [20]. Privacy is a constant interaction where information flows between parties [20, 26]. Privacy expectations vary based on context changes.

The Approximate Information Flow (AIF) [20] proposes a model which deals with situations where some actors hold private information relevant to everyone. These situations happen in special environments called environments with asymmetric information: the flow of information from data owners to data users is more important than from data users to data owners creating asymmetry. For instance, when users browse the Web, information is continuously collected by third-parties to build accurate user profiles but users have few occasions to find out which information has been collected. The goal is to end up with minimum asymmetry – the principle of minimum asymmetry:

"a privacy-aware system should minimize the asymmetry of information between data owners and data users, by:

- Decreasing the flow of information from data owners to data collectors and users.
- Increasing the flow of information from data collectors and users back to data owners" [20].

Even though recovery and response tools are not explicitly mentioned, this model takes into account the lifecycle of privacy data that is: collection, access and second (and further) use. This is a reminder that controlling "second use of personal data in the general case is very hard" [20]. Actual prevention, avoidance or detection tools can do little against the second use of information. In online scenarios, detection technologies are missing. Indeed, this lack of detection makes reacting, recovering and sanctioning difficult.

Although there are different tools and technologies available to prevent attacks on privacy when online applications are used, few tools are available for detection of actions that violate privacy agreements. In the remaining of this paper, we describe such a tool for detection of second use of private information. Our tool detects exchanges of subscribers' private email addresses between online providers and checks whether or not those providers have respected their advertised privacy policy. In doing so, the flow of information from data collectors and users to data owners is increased. When users wonder if they should register to providers due to the risks generated by second use of their private information, they can consult our system to check how well those providers deal with second use of email addresses. Based on this information, users may avoid registering to providers leaking private information such as emails. The conclusion depicts future work that could be undertaken on top of the tool described in this paper.

2. PR3 Email Honeypot

At the beginning of this section, the approach used for building our tool for detection of privacy leakage occurring when emails are given to online providers is described. Then, we detail the design and implementation of this tool. The end of this section briefly surveys related work.

2.1. Approach

PRoactive PRivacy PRotection (PR3) is an approach for privacy protection mechanisms. The tool, described in the next subsection, is the first step to make our PR3 approach real. Other tools will be created in order to be able to publicly publish this approach in details. The main idea behind this approach is that users can also use the technological infrastructure and act actively against potential attackers. Users can act and not only be acted upon. This is not really a new category of privacy protection mechanisms because PR3 can be applied to different categories, such as prevention or detection. It is not only targeted for new environments such as pervasive ones but also for traditional online environments. Actually, some past PETs can be seen as compliant to the PR3 approach. Nevertheless, generally, actual privacy protection mechanisms can be categorized as passive defensive mechanisms: legislation, P3P, encryption, pseudonymity and anonymity. PR3 mechanisms are used to actively protect privacy. The old adage says "the best defense is a good offense". However, rather than going straight for the attack, the new mechanisms would proactively defend privacy.

The PR3 approach emphasizes that detection is important because users must be in control. A PR3 system should proactively search for what is known on the profile to increase the flow of information from data collectors and users to data owners. The same channels used by marketers and consumers of profiles should be used. Second use of private data could be detected thanks to backward liability: if data collectors hold private data that they are not allowed to hold, they should be liable if they cannot point to the originator of this data and so on. Another trick would be to use the notion of "honeypot" [30]. Easy detectable private data would be sent through the privacy leaking channels: this would ease the work of privacy detective agents.

In fact, the concept of honeypot is used in our tool. Spitzner's honeynets "are nothing more than a type of honeypot, which is a security resource whose value lies in being probed, attacked, or compromised. Conceptually, honeypots are simple. You create a resource that has no production value or authorized activity. This means if any packet or any interaction is attempted with your honeypot, it's most likely a probe, scan, or attack" [29]. The goal of our tool is to detect whether providers asking for emails at the time of registration respect their privacy policies regarding the use of that email address. It is not detection per se – detection of second use of real private data – because the data is just decoy information. It is why it differs from how detection is traditionally seen – active search of second use of real private data – in the sense that it tries to assess the provider before giving real data. Moreover, the results help to increase awareness and avoidance because the user can choose to give their data based on that supplementary information. So, the central idea is to provide an email address as a security resource with which any future interaction would be monitored. Also, each email address provided is unique and only disclosed once to the online provider at the time of registration. Our email addresses cannot be found on Web pages or newsgroups. By analyzing future interactions, it is possible to infer whether the privacy policy opted-in at the time of registration has been respected or not. Hence, it slightly differs from Spitzner's view because some interactions with the specific email addresses can be legitimate depending on the opted-in privacy policy.

Another goal of the project is give users the ability to determine whether a provider respects its privacy policy or not. The alternative would be to have a list of all providers and their results concerning their respect of privacy policies. The latter list could be consulted by users to find out if the new provider that they want to interact with, respects what it claims concerning privacy protection. Nevertheless, such a list raises legal concerns because the publisher of the list may be sued for "not strictly proven" information about well-known providers unhappy to see their name and their results on this list. A second benefit of allowing users to insert their providers of interest comes from the fact that it is very time-consuming to manually register providers. A fully automated process cannot be envisaged since many providers forbid such automated process by including hard challenges to be responded by computers without the help of humans [1]. It is why the system is designed to be used by many users called "contributors". As an aside, by choosing contributors with different profiles (gender, interests etc), the coverage of providers inserted into the system is more likely to contain the most representative providers for any user. Obviously, by distributing the work among contributors, the system can reach a high number of registered providers in a more expedient manner.

Therefore, the concept of honeypot is applied to the email subscription process. Each time one of the contributor of the projects subscribe to one of the numerous online providers, a decoy email address associated with a decoy user account is created and given to the provider to complete the registration. Thus, the provider only knows this email address. Later, if emails from other providers are sent to this email address, there is a high probability that the email address has been exchanged between the providers involved. Along the way, the system will track down exchanges of subscribers' private emails between online providers and will provide interesting results, such as which providers respected their privacy policies and how often private email leakages occur. Figure 1 is a sequence diagram depicting a simple scenario:

- A contributor registers a new decoy user to provider 1.
- At some stage provider 1 sends an email to the unique email address given at time of registration.
- The opted-in privacy policy specifies that emails sent by provider 1 are legitimate (the next subsection explains how the check of privacy policy is achieved).
- Later, provider 2 sends an email to the same unique email address: the chance is high that provider 1 gave the unique email address to provider 2.
- This specific privacy policy is suspected to have been breached because this policy claimed that provider 1 would keep the unique email address for its own use.

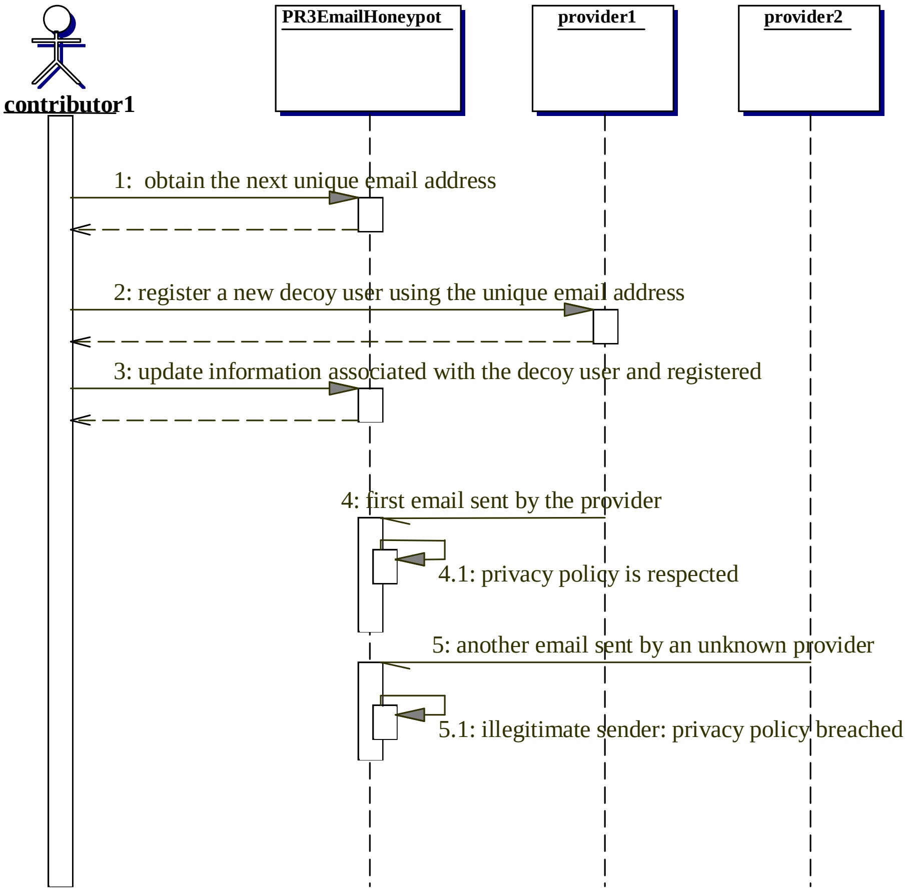
contributor1
PR3EmailHoneypot
provider1
provider2
1: obtain the next unique email address
2: register a new decoy user using the unique email address
3: update information associated with the decoy user and registered
4: first email sent by the provider
4.1: privacy policy is respected
5: another email sent by an unknown provider
5.1: illegitimate sender: privacy policy breached

Figure 1: Sequence diagram of a simple scenario

2.2. Implementation Overview and Lessons Learnt

At the heart of the system there is a mail server that has been modified in order to create decoy email addresses on demand. The Java Apache Mail Enterprise Server (JAMES) [19] has been used to achieve that: it provides an SMTP server and a Java API, called the mailet API, to write Java code to process incoming email messages. Figure 2 depicts the different components of the system.

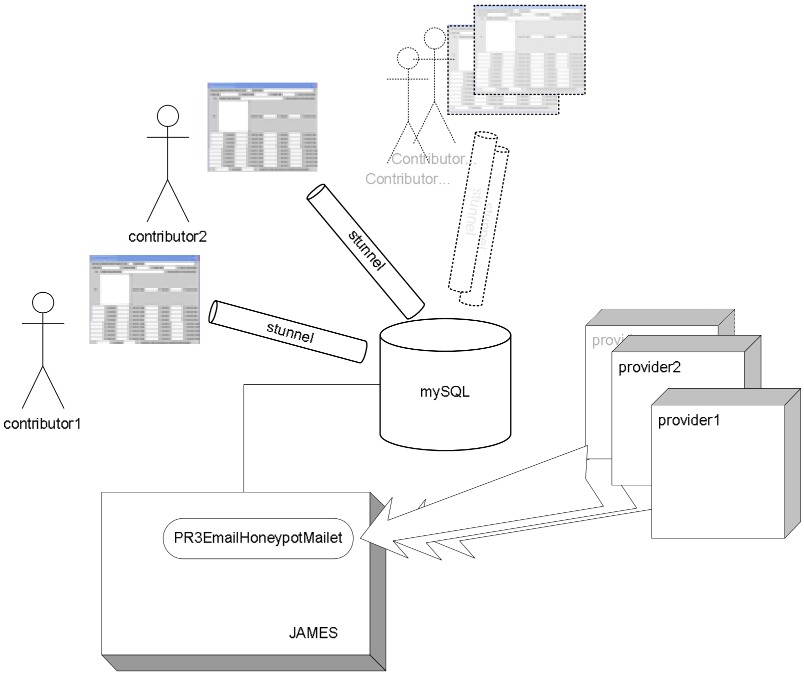

Figure 2: PR3EmailHoneypot main components

A specific mailet, called the PR3EmailHoneypotMailet, has been programmed to obtain a first guess on whether the opted-in privacy policy has been respected or not. JAMES is connected to a MySQL database, which contains tables to manage the following main types of information:

1. Provider information: name identifier, domain name, URL, type of provider (e.g. federated identity management provider), the date and time of creation.
2. Decoy user information: unique email address of the decoy user, the name identifier of the provider, the date and time of creation, a boolean indicating if the user has opted-in for the privacy policy, the current number of emails received, and a list of 2-tuple (information type, information value), which might be used to check if emails are sent based on the decoy user profile.
3. Privacy policy information: the name identifier of the provider, a boolean indicating whether the provider claims having a so-called privacy policy, the type of the policy, which is a summary of the full privacy policy copied into another field.
4. Other custom information: tables used to store the first guess on whether the emails received are compliant with opted-in privacy policies or not.
5. Standard JAMES information: all the emails received are permanently stored anyway in the standard JAMES folders (inbox, deadletter, etc.).

Since all emails received are stored, it will be possible to reuse the input received by the PR3EmailHoneypot for off-line sorting out of the emails in case better sorting algorithms are available. At the time of writing, an email is considered to comply with the privacy policy of the provider by comparing which source sent the email with which source is allowed by the privacy policy to send an email to the specific address. There are four types of privacy policies summarized below:

1. “No forward to partners at all, but direct email okay SIG:PP0”: At the time of registration, the decoy user agreed to receive emails from the provider but from no other providers. The provider is considered the source of an email when an occurrence of the name, domain name or URL is found in the email (sender email address, subject or body of the mail). After publishing the latter, it would be easy to

thwart this mechanism by randomly adding names of well-known trustworthy providers in the message. Nevertheless, before publication, it was considered applicable. Also, another process could be used for off-line processing of the input received by the PR3EmailHoneypot. The online process only gives a dynamic first guess of how many providers have broken their privacy policy to date. If an email is received at this unique email address from a provider different than the one linked to this email address, the count of providers suspected to have breached their privacy policy is increased by one.

2. “Forward to all partners SIG:PP2”: When this privacy policy has been opted-in, any email arriving at the associated unique email address must be considered legitimate.
3. “Forward to restricted list of partners only SIG:PP1”: For this category of privacy policy, it is currently difficult to know which sender is legitimate or not because most of the time the list of partners is unknown. The dynamic sorting algorithm of the PR3EmailHoneypot places emails received in such case into a “not sure” list if the sender is not guessed as the source provider.
4. “Don't know, too complex, no time SIG:PP-1”: Contributors should sparingly use this type of policy because it means that the decoy user opted for a privacy policy which is too complex to be placed into one of the previous ones. The dynamic sorting algorithm of the PR3EmailHoneypot puts all emails received in such case into the “not sure” list.

Thus, it is difficult to find out whether an email comes from a specific provider or not and to understand quickly very long privacy policies. We argue that the difficulties encountered could be partly answered by the set of the following guidelines:

- The privacy policy may specify which unique email address will be used for future contact. It may not be a problem for companies to choose one specific email address for all future contacts and enforce its use throughout the company and the online services used. It would even be more efficient to specify a mechanism (e.g. a public key certificate) along with that email address in order to be able to rigorously authenticate the sender, as Tompkins and Handley [32] have recently suggested.
- If the privacy policy says that a restricted list of partners will obtain the email address in question, the list of unique email addresses potentially used should be specified for the same reasons as in the previous guideline.
- A one-sentence summary of what the privacy policy allows would be clearer than very long privacy policies. However, we are aware of the difficulties of writing clear statements due to the complexities of legal contracts.

Each contributor receives a Java application securely connected through a secure tunnel to the database. That application launches a GUI which returns the next unique email address. The GUI also contains different fields that the contributor can fill according to what is filled during the online registration process of the new decoy user with the provider (provider, privacy policy and decoy user information). Then, the contributor commits the new information into the database and can finish the registration process with the provider by giving the unique email address associated with the new decoy user. Nevertheless, the work of the contributor is not finished yet because some providers require an extra step to confirm the registration (e.g. the first email sent contains a Web link that once clicked confirm the registration). Currently, there are two mechanisms used to allow the completion of the registration process in spite of a confirmation email:

1. Any first email sent to a unique email address is forwarded to the contributor that registered the decoy associated user. The contributor can carry out the remaining required operations detailed in the response message to complete registration.

2. Any email containing one of the words present in a local list is also forwarded to the contributor that registered the decoy user associated. At the moment, the word “confirm” is the only word present in this list. Obviously, after the publication of this paper, it will be easy to misuse this feature by adding that particular word in any email causing excessive processing overhead.

In JAMES, every incoming email message goes through a number of mailets according to which mailet and in the order it has been configured in a static XML configuration file. In the current configuration of the system, after being processed by the PR3EmailHoneypotMailet, the emails are not forwarded – the mail server cannot be used as an open relay. Nevertheless, at the end of the process, another custom mailet has been programmed to send a warning message to the JAMES administrator each time a certain number of errors and guesses of breaching of privacy policies is reached.

There are some limitations concerning the level of secrecy of the unique email address. The unique email address is only divulged to the target provider but it is still possible to obtain the unique email address:

- By launching a brute-force attack on the mail server by trying any combination of characters possible. Three partial counter-measures have been thought to minimize this limitation though. If the domain name of the mail server is not published on the Web, it is more unlikely that an attacker would target this domain. Also, each unique email address is quite long and complicated in order to keep the entropy high. In the current implementation, the email starts with a real first name (to masquerade a real person’s email) followed by the first 5 characters of the provider name then a number between 0 and 1000. A scheme that provides higher entropy may be adopted at a later stage. If the brute-force attack is detected (e.g. many unknown unique email addresses are tried during a short period of time), the emails related to this attack are not taken into account. Also, the attacker cannot guess whether the email addresses are valid because no reply is sent.
- By obtaining the email address while it is in transit on the Internet due to eavesdropping. This type of attack is difficult to achieve at key Internet nodes because they are well protected, but it is feasible on some LANs. Again, a flow of emails to many unique email addresses could be inferred as suspect because usually a small number of decoy users are registered per provider. Such emails could be discarded.

These limitations must be addressed and some better “counter-measures” should be developed. For example, if verifiable keys could be exchanged during the registration as mentioned in the above set of guidelines, it would be obvious whether the provider has divulged the secret or not. The disclosure could be even unwanted by the provider in case of an attacker who would successfully steal the information from the provider’s database. Nevertheless, the provider might still be responsible: depending on the laws, in some cases, the providers must make sure the data can not be stolen.

The first version of the PR3EmailHoneypot was deployed at the end of May 2003.

2.3 Related Work

There are several organizations providing privacy assurance services. These organizations verify that the providers meet their proprietary program’s core requirements (e.g. for data gathering and dissemination) and allow the successful providers to display their assurance seal [3, 33, 34]. Users may choose to avoid registration based on the fact that providers do not display theses seals. Nevertheless, these organizations do not detect if leakages occurred. Also, they only carry out audits from time to time while the PR3EmailHoneypot is constantly listening.

Since spammers are keen on using open relay mail servers [22], there have been various attempts to set up open relays as honeypots in order to learn spammer techniques. As far as we know, there is no major project involving open relay honeypots in operational state. The main difficulties come from the fact that if emails are really forwarded, mail servers' owners may be liable, and it also aggravates the number of spam emails. Conversely, if incoming emails are not forwarded, spammers have means to detect that occurrence and stop using the honeypot relay.

An approach that focuses on profiling spammers is being developed by John T. Draper [14]. Using the Crunchbox security system, a series of honeypot mailboxes have been set up that are used to poison the email address databases of spammers. The system allows the tracking of spam email messages as they arrive, facilitating the studying of patterns related to spammers. Moreover, the received spam messages are used to dynamically create detection rules for the Snort intrusion detection system, which are triggered when similar spam messages arrive in the monitored network. This allows the almost real-time contact of the ISP that is used by the spammer and presents the opportunity of taking effective countermeasures.

Different techniques are used to avoid, detect and harass spambots, also called email harvesters − programs that extract email addresses from Web pages. The extracted emails are then used as targets for spam [24]. Spambots can be stopped using traps e.g. based on IPchains, IPtables, Apache…[17, 18, 25]. For instance, using Apache, by embedding a decoy email address along with date of access and host IP address in each page served, spam may be traced back to hosts or robots. Other tools try to "poison" spambots by feeding them with decoy data [6].

3. Conclusion and Future Work

When people receive unsolicited emails, it is also a loss of privacy – the right "to be left alone" is not respected. However, spam can also be used to detect providers, e.g. e-commerce Websites, which default on their privacy policy by giving away private information, such as email addresses. If spam is received by an email address that has only been given to a single provider, we may conclude that this provider has either given the address to the spammer or very poor security. In either case, registration with that provider should be avoided.

The concept of honeypot has been successful in other domains to lure attackers and learn their techniques and behaviours. In this paper, the honeypot concept is applied to the email system in order to proactively detect "bad" providers (i.e. those who do not respect their privacy policy) where the honey consists of unique email addresses. It is called proactive detection because it aims at knowing the trustworthiness of providers before actually starting the collaboration. That approach is in line with the PR3 approach that we are proposing for privacy protection. PR3 highlights the fact that users can also act on the system and not only passively protect themselves from attackers being acted upon. Proactive searching of the leakage of private data deals with second and further uses. In our case, it is even proactive detection because the system does not detect whether real private information has been leaked but instead detects decoy information. The results obtained should also increase awareness and avoidance. In doing so, the flow of information from data collectors and users to data owners is increased. When users wonder if they should register to providers due to the risks generated by second use of their private information, they can consult our system to check how well those providers deal with second use of email addresses. Based on this information, users may avoid registering to providers leaking private information such as emails. This tool is the first step to make our PR3 approach real. Other tools will be created in order to be able to publicly publish this approach in detail.

In this paper, we present the current implementation of the tool called PR3EmailHoneypot used for detection of second use of private information. Although there are different tools and technologies available to prevent attacks on privacy when online applications are used, few tools are available for detection of actions that violate privacy agreements. Our tool detects exchanges of subscribers' private email addresses between online providers and checks whether or not those providers have respected their advertised privacy policy. The number of providers listed in the database should cover the most common online providers since the tool is shared among different contributors who manually insert providers when they have the opportunity to do so. The project goal is to increase in size in terms of distribution by connecting different honeypots of this type together and study what can be improved and obtained. Other types of distributed honeypots can be used for prediction and early warning [29]. The publication of this paper may facilitate getting more people involved and turn the PR3EmailHoneypot into a community effort. However, there are limitations in the actual implementation. An automated registration may be used for registrations that do not use challenges, which is difficult for computers to solve without human intervention. Actually, results are more guesses than fully provable facts. Some guidelines are able to get provable results even though it requires modifying the content of privacy policies and the registration process. It would also be helpful to be able to dynamically update the list of privacy policy summaries and the set of processes used to judge if the senders are who they claim to be. Eventually better offline processes may be reapplied to the list of emails to improve the confidence in identifying providers who breached their privacy policy.

Long-term results by the current implementation may show hidden partnerships between providers exchanging email addresses but it is not known if such results could be published, without legal worries.

4. Acknowledgments

We are grateful to the JAMES developers community [19] and the members of the Trinity College Dublin Security Interest Group (TCD-SIG) [31].

5. References

[1] L. v. Ahn, M. Blum, N. J. Hopper, and J. Langford, "CAPTCHA: Using Hard AI Problems For Security", in *Proceedings of Eurocrypt'03 International Conference on the Theory and Applications of Cryptographic Techniques, LNCS 2656*, pp. 294-311, Springer-Verlag, Berlin Heidelberg, 2003, http://www-2.cs.cmu.edu/~biglou/captcha_crypt.pdf.

[2] T. Aura and C. Ellison, "Privacy and Accountability in Certificate Systems", Research Report A61, Helsinki University of Technology, 2000, http://www.tcs.hut.fi/old/papers/aura/HUT-TCS-A61.pdf.

[3] BBBOnline, http://www.bbbonline.org/privacy.

[4] O. Berthold and H. Langos, "Dummy traffic against long term intersection attacks", http://page.inf.fu-berlin.de/~berthold/publ/BeLa_02.pdf.

[5] B. D. Brunk, "Understanding the Privacy Space", in *First Monday*, vol. 7, no. 10, Library of the University of Illinois, Chicago, 2002, http://www.firstmonday.org/issues/issue7_10/brunk/index.html.

[6] D. Carraway, "Sugarplum - spam poison", Website, 2003, http://www.devin.com/sugarplum/.

[7] M. Casassa Mont, P. Bramhall, M. Gittler, J. Pato, and O. Rees, "Identity Management : a Key e-Business Enabler", Technical Report HPL-2002-164, Hewlett-Packard, 2002, http://www.hpl.hp.com/techreports/2002/HPL-2002-164.pdf.

[8] D. Chaum, "Achieving Electronic Privacy", in *Scientific American*, vol. August, pp. 96-100, 1992, http://www.chaum.com/articles/Achieving_Electronic_Privacy.htm.

[9] D. Chaum, "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms", in *Communications of the ACM*, vol. 24 (2), 1981, http://world.std.com/~franl/crypto/chaum-acm-1981.html.

[10] T. M. Cooley, "A Treatise on the Law of Torts", Callaghan, Chicago, 1888.

[11] L. Cranor, M. Langheinrich, M. Marchiori, and J. Reagle, "The platform for privacy preferences 1.0 (P3P1.0) specification", W3C Recommendation, 2002, www.w3.org/TR/P3P/.

[12] DataProtectionWorkingParty, "Opinion 10/2001 on the need for a balanced approach in the fight against terrorism", The European Commission, 2001, http://europa.eu.int/comm/internal_market/en/dataprot/wpdocs/wp53en.pdf.

[13] DataProtectionWorkingParty, "Privacy on the Internet - An integrated EU Approach to On-line Data Protection -", The European Commission, 2000, http://europa.eu.int/comm/internal_market/en/dataprot/wpdocs/wp37en.pdf.

[14] J. T. Draper, "Following their patterns", in *Proceedings of Spam Conference*, Cambridge, MA, USA, 2003, http://www.spamconference.org.

[15] EU, "Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data", 1995, http://europa.eu.int/ISPO/legal/en/dataprot/directiv/directiv.html.

[16] EU, "Directive 2002/58/EC of The European Parliament and the Council of the European Union", in *Official Journal of the European Communities*, 2002, http://www.oftel.gov.uk/ind_info/eu_directives/data0702.pdf.

[17] N. Gunton, "Stopping Spambots: A Spambot Trap", Website, 2003, http://www.neilgunton.com/spambot_trap/.

[18] L. T. Hughes, "Defending Against Email Harvesters, Leechers and Web Beacons", Website, 2003, http://linux.oldcrank.com/tips/antibot/.

[19] JAMES, "JAMES: Java Apache Mail Enterprise Server", Website, http://james.apache.org.

[20] X. Jiang, J. I. Hong, and J. A. Landay, "Approximate Information Flows: Socially Based Modeling of Privacy in Ubiquitous Computing", in *Proceedings of the 4th International Conference on Ubiquitous Computing (Ubicomp 2002), LNCS 2498*, pp. 176-193, Springer-Verlag, Berlin Heidelberg, 2002, http://guir.berkeley.edu/projects/uisper/pubs/ubicomp2002-aif.pdf.

[21] L. Lessig, "The Architecture of Privacy", in *Taiwan Net'98 Conference*, 1998, http://cyberlaw.stanford.edu/lessig/content/articles/works/architecture_priv.pdf.

[22] G. Linberg, "RFC 2505 Best Current Practice", Network Working Group, 1999, http://www.networksorcery.com/enp/rfc/rfc2505.txt.

[23] D. M. J. Martin, R. M. Smith, M. Brittain, I. Fetch, and H. Wu, "The Privacy Practices of Web Browser Extensions", in *Communications of the ACM*, vol. 44(2), pp. 45-50, 2001, http://citeseer.nj.nec.com/447541.html.

[24] G. S. Mullane, "Spambot Beware", Website, 2003, http://www.turnstep.com/Spambot/index.html.

[25] R. Nilsson, "Guestbooks: Protecting Email Addresses", Website, 2003, http://watersgulch.com/digital/articles/gbook_protectn.html.

[26] E. M. Noam, "Privacy and Self-Regulation: Markets for Electronic Privacy", 1997, http://www.ntia.doc.gov/reports/privacy/selfreg1.htm#1B.

[27] S. Pearson, "Trusted Agents that Enhance User Privacy by Self- Profiling", Technical Report HPL-2002-196, Hewlett-Packard, 2002, http://www.hpl.hp.com/techreports/2002/HPL-2002-196.html.

[28] M. K. Reiter and A. D. Rubin, "Anonymity Loves Company: Anonymous Web Transactions with Crowds", 1999, http://citeseer.nj.nec.com/reiter99anonymity.html.

[29] L. Spitzner, "The Honeynet Project: Trapping the Hackers", in *IEEE Security & Privacy*, vol. March/April, 2003, http://computer.org/security/v1n2/j2spi.htm.

[30] L. Spitzner, "Honeypots: Tracking Hackers", in *ISBN 0321108957*, Addison-Wesley, 2002.

[31] TCD-SIG, "TCD SIG: Trinity College Dublin Security Interest Group", Website, http://www.cs.tcd.ie/Jean-Marc.Seigneur/tcdsig/.

[32] T. Tompkins and D. Handley, "Giving E-mail Back to Users: Using Digital Signatures to Solve the Spam Problem", in *First Monday*, vol. 8, no. 9, Library of the University of Illinois, Chicago, 2003, http://firstmonday.org/issues/issue8_9/tompkins/.

[33] TRUSTe, Website, http://www.truste.org.

[34] WebTrust, http://www.cpawebtrust.org.