Private incident reporting using onion networks

Evangelos Katsadouros[1], Christos Chatzigeorgiou[1][0000−0003−2486−9897], Michalis Feidakis[1][0000−0002−6412−8420], Dimitrios G. Kogias[2][0000−0001−8985−6136], and Charalampos Z. Patrikakis[1][0000−0003−1921−4466]

[1] Department of Electrical and Electronics Engineering, University of West Attica, Egaleo, Greece
{katsadouros.v, chrihatz, m.feidakis, bpatr}@uniwa.gr
[2] Technical Department, iTrack Services Ltd., Piraeus, Greece
dimikog@itrack.gr

Abstract. Privacy mechanisms in internet have bothered the researchers and the users since the beginning of it. In an era where everything is done from a mobile phone, users have trust issues about reporting incidents in authorities without revealing their identity. This paper presents an architecture that helps users to report malicious events in the authorities using their smartphones, while ensuring their identity will not be exposed.

Keywords: Private communication · Onion network · Incident reporting.

1 Introduction

In recent years, people have become more aware of their own data's privacy, resulting from increasing data breaches in government organisations and corporations and surveillance in communications. *Privacy Enhancing Technologies - PETs* have been emerged towards the protection of privacy in Information and Communication Technologies (ICT). A formal definition of PETs has been given by Borking and Raab as *"a coherent system of ICT measures that protects privacy by eliminating or reducing personal data or by preventing unnecessary and/or undesired processing of personal data; all without losing the functionality of the data system"* [1, p. 2].

PETs can include various methods including anonymization, psedonymization and cryptography. According to Pfitzman and Hansen, *"anonymity of a subject means that the subject is not identifiable within a set of subjects, the anonymity set"* and *"a pseudonym is an identifier of a subject other than one of the subject's real names"* [2, p. 9, p. 21]. Finally, cryptology refers to techniques for communicating securely in the presence of adversaries [3].

Privacy can be applied in many everyday communications like e-mail, instant messages and Internet browsing. Even in deep learning, researchers have applied techniques to protect user data [4], [5], [6].

Crime and incident reporting constitutes an important situation that requires privacy. People tend not to report crimes, as they are afraid of their identity being somehow exposed to the criminal. Similarly, they prefer only to report the incident without having to testify to the law enforcing officers [7].

To this end, we propose a solution that allows anonymous incident reporting through an onion network, like Tor [8]. More specifically, users will be able to report incidents in a local area, using an application in their mobile phone. Receivers could be any law enforcement agencies, security companies or even municipal authorities in charge. Reports will pass through the onion network, make them encrypted and difficult to trace them back to their sender. Using this solution, users will be able to report incidents more easily and without the concerns discussed in the previous paragraph.

An implementation of the proposed solution has been designed in DESMOS project for smart and inter-connected cities [9]. The visitors of Trikala city in Greece, will have the chance to report incidents (e.g., thefts, vandalism) in real time, ensuring their privacy through *onion reporting*. Moreover, the solution can be implemented in similar platforms (i.e., platforms that feature incident reporting), resulting in more incidents being reported and more timely resolve.

The rest of the paper is organized as follows: In Section 2, notable PETs are reviewed. Section 3 describes and analyzes a solution for private incident reporting. Finally, Section 4, includes the discussion for the proposed system.

2 Notable privacy mechanisms

While there have been almost 30 years since Internet's "Hello World" [10], the need for privacy in electronic communications is much longer. During the 80s', PETs were designed to be used in e-mail exchange. There are many reviews about the past and present technologies used for privacy protection and their issues [11], [12], [13].

The first research in PETs was done by Chaum in 1981. He described the "mix" node; a server that hides the correspondences between its input and output messages in a cryptographically strong way [14]. This technique uses public key cryptography to hide the identity of an e-mail sender. Moreover, the recipient's address is hidden from someone who observes the communication channel. Using cryptography, it allows the content of the message to be hidden. Also, this technique allows the receiver to reply to the sender. The "mix" node isn't required to be a universally trusted authority.

In the early 1990s, the rise of the Internet resulted in the increased need to protect the content and the sender of e-mail messages. Mail servers designed to send e-mail messages without identifying the sender – known as *remailers* – started to appear in large scale. The remailers have been further classified in two types, (i) the pseudonymous and (ii) the anonymous.

In the first type, the server sends the message with the sender's address replaced by a "pseudoaddress". This allows the recipient to reply to the message through the remailer. Penet remailer, by Johan "Julf" Helsingius, was one of

the most known psedonymous remailers which operated in Finland from 1993 to 1996 [15]. However, the database which mapped the sender with a pseudonym was saved on the same server as the remailer. This made the server vulnerable to attacks, which led to server shutdown.

Anonymous remailers on the other hand, remove any information that can lead back to the sender, and then forward the e-mail to the recipient – Type I (or Cypherpunk) remailers. They were developed by Eric Hughes and Hal Finney [16]. The remailer discards the original mail headers and adds new before sending the message. Moreover, the mail from the sender is encrypted and decrypted in the server. Also, there is the option to chain the message through multiple remailers. The problem in this type of remailers is that the size of the message is the same from the sender to the recipient, thus the message can be tracked in the network just by observing its length.

Type II (or Mixmaster) remailers were developed by Lance Cottrell in order to address the traffic analysis issue in Type I remailers [17]. This was achieved by using fixed-size packets. Also, the messages were sent with delay from the remailer. However, in case of few incoming messages, dummy traffic must be generated to avoid the analysis. Also, as with Cypherpunks, the reply is not possible, unless a reply address is included in the message.

In 2003, Danezis et al., proposed the Type III (or Mixminion remailers) [18], aiming to solve the issues of Mixmaster. A network of nodes is used which receive, decrypt, re-order and re-transmit the messages. The Mixminion breaks each mail in equally sized packets and then sends it through the network until it reaches its final destination. A great improvement over Type II remailers is the ability to reply to e-mails.

Based on the mix networks, Syverson et al., presented the *onion routing* in 1997 [19]. In their proposal, the message to be transmitted, opens a circuit on the network from which each packet is sent. When the message is transmitted, the circuit closes. In the second generation, known as *Tor (The Onion Router)*, a traditional network architecture is used where a directory service hosts a list of the volunteer servers [8]. The client downloads the list and chooses 3 random nodes from which the message is relayed. The client then connects to the first node and requests to the first node to connect to the next one. In each channel, a Diffie-Hellman key exchange is performed.

An architecture for reporting incidents while preserving users' privacy was proposed in 2017 [20]. The approach consists of two servers with unique role. The first server, named Privacy Protection Proxy Server, is used to remove any user identifiable information from the message, while the Control Centre Server is used for decrypting and forwarding the message to its destination.

3 Onion Reporting

The proposed anonymization network ensures user's anonymity on sending reports. This solution is based on the Tor project and on "Ensuring Anonymity For Incident Reporting by Utilizing Onion Networks" [21]. The main aspects of

this solution are the unidirectional communication – from user to first responder team, the use of nested cryptography among packets and the use of trusted nodes in the network. Furthermore, the final receivers are trusted public mechanisms like police, fire department, etc.

The anonymization network consists of five different parts: (i) Client, (ii) Directory Server, (iii) Relays Manager, (iv) Trusted Node and (v) Trusted Receiver which is the final destination of the message.

– The Client is the application that the user is going to use to make an anonymized report.

– The Directory Server maintains a list of all nodes in the network and their status. Also, it acts as a registration endpoint for each new node in the network.

– The Trusted Node is a service, hosted by a trusted governance mechanism, that runs software which handles the packets. In specific, the Trust Node is capable of:
  - receiving packets from other nodes
  - receiving packets from end users
  - decrypting a packet
  - forwarding packets to other nodes or to the endpoint
  - informing the "Directory Server" for its creation
  - creating unique keys using Diffie-Hellman [22]

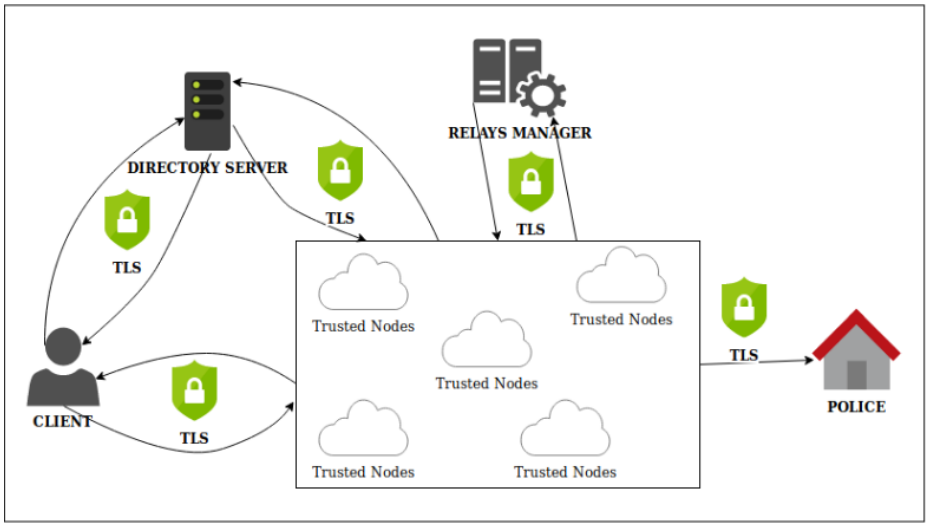

Fig. 1. Anonymization network architecture

– Relays Manager is a service hosted by governance mechanisms to manage the nodes that provides to the anonymization network. Its main tasks are managing the nodes' up time, the creation of new node, and the destruction of existing node. When existing nodes are destroyed, their log information are destroyed as well, thus, enhancing users' privacy.

– Finally, the Trusted Receiver is the endpoint of the network. The endpoints are in some way static because they're limited to governance mechanisms.

Figure 1 depicts the architecture of the anonymization network along with the communication relationships among its parts. The Client communicates only with the Trusted Nodes and the Directory Server, but never directly with a Trusted Receiver (e.g. the police). The Client, also, communicates with the Directory Server to retrieve a list of all active nodes, creating the path to the endpoint. The path is a set of three different and randomly selected Trusted Nodes. Moreover, before the Client sends a report, it negotiates the secret keys with each different node. Each Trusted Node communicates with the Directory Server, notifying both its creation and destruction. Finally, the Relay Manager manages all the Trusted Nodes in the cluster.

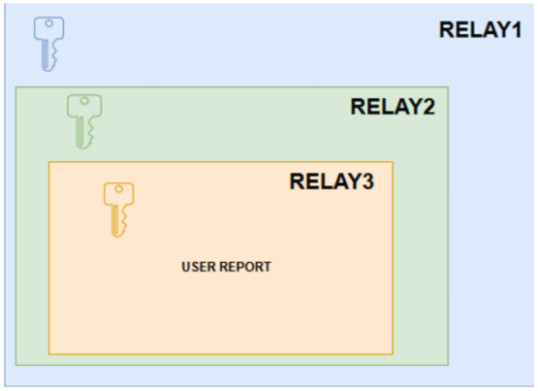

Fig. 2. Anonymization network packet

To secure the communication in the network, all of its parts use TLS v1.2, to ensure the confidentiality and integrity of communication. Also, the client creates unique secret keys with each node. Cryptography is used in the transport layer (TLS) and in the application layer (AES). The two parts of the network negotiate the private key (application layer) in a secure way using the Diffie-Hellman protocol. To send a report, the client chooses randomly three different nodes of the network. Using the negotiated keys, the client encrypts the final packet three times (Figure 2). These keys are used only for one report. After sending the report, the keys are destroyed in both the client and the node.

Finally, random delay is inserted between packet transmission to prevent traffic-analysis attacks.

The final packet is a 3-layer packet –one layer for each node– encrypted with the appropriate key. Each node receives the packet and decrypts it with the key that has exchanged with the client. Inside each packet is information about the next destination, the data that the node has to forward, the type of the packet, a unique id related to the private key, and a flag of exit node if the node is the last one that has to provide the packet to Trusted Destination. The packets are transferred using the JSON format. JSON was preferred over other data interchange formats (e.g., XML) because it is more lightweight and it is easier to integrate more Trusted Receivers, since it is widely used.

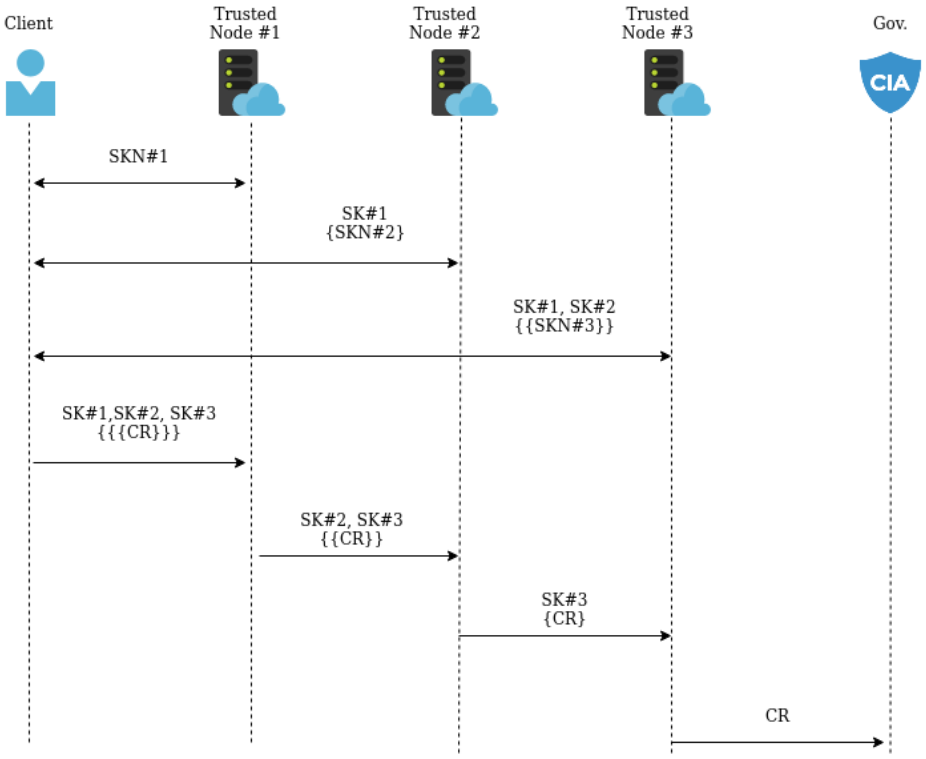

Fig. 3. Report Flow in anonymization network

Figure 3 depicts the communication flow in the anonymization network for sending a report. After the random creation of the network path, the client has to negotiate the private keys with each Trusted Node of the network. The communication flow begins between the client and the first node. Using Diffie-Hellman key exchange, they negotiate a unique private key. Afterwards, the client communicates with the next node to create a unique key, while the packet

is a 2-layer packet encrypted with the secret key negotiated with the first node. In the same way, the client negotiates a key with the third node, by sending a 3-layer packet encrypted this time with the secret keys of the first and the second node.

By collecting three secret keys, the client is ready to send the report. It creates a 3-layer packet, three times encrypted with the secret keys negotiated with the aforementioned nodes. The root of this packet is the user's report. When the packet is created, the client sends it to the first node. The first node decrypts the packet with the appropriate key, reads the route information and forwards the 2-layer packet to the next node. The same procedure is followed from the second node of the path. Finally, the third node, decrypts the packet and, using the information inside, sends the packet to the appropriate destination. Due to unidirectional communication and the lack of nodes recording route information, communication for the destination to the client is not possible.

4 Conclusions

The paper presented an approach for private and safe reporting of incidents in various authorities. The presented solution makes it almost impossible for third parties to read or tamper the report, except from the receiver the report is sent to. An attacker would have to decrypt three layers of encrypted data in order to gain access. Time correlation is avoided due to the time delay that each node randomly adds between the arrival and the departure of each packet. The receiver of the report is not able to find its original sender, since it was not send by the sender, but by three irrelevant nodes instead.

In contrast with the solution in [20], in our approach, the incident report passes through an additional node which adds an extra layer of security. Moreover, the nodes are different and selected randomly for each report, which makes more difficult for an attacker to track reports from a certain user.

One limitation is that in the proposed architecture, it is not possible to send replies to the sender, as nodes don't know how to get back to the sender. Moreover, multiple nodes are required to function properly, otherwise, the route will be always the same, making it easy to find the original sender

Acknowledgments

This research has been co-financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH-CREATE-INNOVATE (project code: T1EDK-03487).

References

1. John Borking and Charles Raab. Laws, pets and other technologies for privacy protection. *Journal of Information, Law and Technology*, 2001, 01 2001.

2. Andreas Pfitzmann and Marit Hansen. A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management. http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.34.pdf, August 2010. v0.34.
3. Ronald L. Rivest. *Cryptography*, volume 1, chapter 13, pages 717–755. Elsevier.
4. H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data, 2016.
5. Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, page 308–318, New York, NY, USA, 2016. Association for Computing Machinery.
6. C. Chatzigeorgiou, P. Kasnesis, and L. G. Toumanidis. Exploiting edge computing for privacy aware tourism demand forecasting. *IT Professional*, 21(3):19–25, May 2019.
7. Samuel L Myers. Why are crimes underreported? what is the crime rate? does it" really" matter? *Social Science Quarterly*, 61(1):23–43, 1980.
8. Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. Technical report, Naval Research Lab Washington DC, 2004.
9. Desmos. http://desmos-project.gr/en/homepage-en. Accessed: 2020-02-25.
10. T.J. Berners-Lee. The world-wide web. *Computer Networks and ISDN Systems*, 25(4):454 – 459, 1992.
11. Herman T. Tavani and James H. Moor. Privacy protection, control of information, and privacy-enhancing technologies. *SIGCAS Comput. Soc.*, 31(1):6–11, March 2001.
12. I. Goldberg, D. Wagner, and E. Brewer. Privacy-enhancing technologies for the internet. In *Proceedings IEEE COMPCON 97. Digest of Papers*, pages 103–109, Feb 1997.
13. D. Chen and H. Zhao. Data security and privacy protection issues in cloud computing. In *2012 International Conference on Computer Science and Electronics Engineering*, volume 1, pages 647–651, March 2012.
14. David L. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM*, 24(2):84–90, February 1981.
15. Sabine Helmers. A brief history of anon. penet. fi-the legendary anonymous remailer. *Computer-Mediated Comunication Magazine*, 4:9, 1997.
16. Sameer Parekh. Prospects for remailers. *First Monday*, 1(2), 1996.
17. Mixmaster.
18. G. Danezis, R. Dingledine, and N. Mathewson. Mixminion: design of a type iii anonymous remailer protocol. In *2003 Symposium on Security and Privacy, 2003.*, pages 2–15, May 2003.
19. P. F. Syverson, D. M. Goldschlag, and M. G. Reed. Anonymous connections and onion routing. In *Proceedings. 1997 IEEE Symposium on Security and Privacy (Cat. No.97CB36097)*, pages 44–54, May 1997.
20. Christos Chatzigeorgiou, Lazaros Toumanidis, Dimitris Kogias, Charalampos Patrikakis, and Eric Jacksch. A communication gateway architecture for ensuring privacy and confidentiality in incident reporting. In *2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA)*, pages 407–411. IEEE, 2017.

21. Evangelos Katsadouros. Ensuring anonymity for incident reporting by utilizing onion networks. Master's thesis, Dept. Digital Systems, University of Piraeus, Piraeus, 2018.
22. W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, November 1976.