Requirements to send unobservable messages accross the internet

Martin Gwerder

Abstract—In this paper we intoduce the MessageVortex prtocol. A message annonymization prtocol based on zero trust and a distributed P2P architecture without central aspects. It scores over existing work by blending its traffic into suitable existing transport protocol thus making it next to impossible to block it without affecting sigificantly normal users of the transport medium. It allows furthermore to a sender to control all aspects such as degree of anonymity, timing, redundancy of the message transport without disclosing any details to the routing or transporting node.

Index Terms—Data privacy, Message systems, Anonymity, Security

1 Introduction

SINCE whistle blower Edward Snowden disclosed documents that prove the fact that global monitoring of internet traffic is done at a large scale. According to these documents (verified by NRC) NSA infiltrated more than 50k computers with malware to collect classified, or personal information. They furthermore infiltrated Telecom-Operators (mainly executed by British GCHQ) such as Belgacom to collect data, and targeted high member of governments even in associated states (such as the Germans president mobile phone). A later published shortened list of "selectors" in Germany showed 68 telephone and fax numbers targeting economy, finance and agricultural parts of the german government.

This work adresses the above mentioned problem of message recording for either real time analysis or later processing. A normal message sent thoughout the internet must, even when perfectly encrypted, disclose at least the recipient to the router transporting a message. Normally the sender is included as return path or the sender of packets in this information as well. This information is valuable as frequency and message size disclose important facts about the association of the involved parties. This meat information may be hidden with our protocol.

Numerous attempts such as in [1], [2], [3], [4], [5], [6] have been made to use relays [7], mixes [7], or DC-related-networks [8] to anonymize message flow. But most of them have problems as they rely at least on the partial trust to the nodes routing the messages, or some central infrastructures [9], [10], [11], [12]. Exit and entry points are important as they may leak information which is otherwise well hidden within the network. Additionally, a dedicated transport protocol is easy to block since their implementation can be easily identified by used ports. Furthermore, all approaches require to have infrastructure with fixed addressing in the internet making owners of participating nodes identifiable and vulnerable.

Attacks are usually done by obtaining traffic flow information either by a third party observer, or by inserting one or more malicious nodes into a network. By analysing timing, message size, and content important facts about the messages may be learned.

All works analysed for this paper introduced a new transport layer solving these problems. In our approach we decouple the routing layer from the transport layer. By doing so we introduce new degrees of complexity to attack scenarios as messages may use any common transport protocol of the used network.

Our work consists of a routing layer which is completely P2P based without any central infrastructure. Any node is a routing node and may be an endpoint. Decoy traffic generation is controlled by the original sender of a message. Even the generating node is unable to differentiate between message and decoy traffic. As transport media we use common, well known store and forward based protocols. By doing so we have no affiliation to the transport layer. Literally any free-mailer email address or chat account may be converted into a transport media for our protocol.

Using the MeassageVortex protocol any device with a latent or permanent connectivity to the internet may act as routing node. It conceals its own traffic with the routed traffic making it harder for any adverser to observe and identify traffic. By applying the zero trust model we give full control of all traffic to the original sender of the message. He controls message flow, redundancy, degree of anonymity, timing, and many more aspects of the message transport throughout the whole network. This is done without disclosing any of these parameters to the participating nodes.

To limit possibilities of DoS within the system and guarantee an efficient handling of messages, MessageVortex nodes rely on unlinked, ephemeral identities which are created in a proof of work system. While it is technically easy to use a MessageVortex node, it is hard to run traditional attacks against them. The amount of work required to disrupt services or do traditional attacks grows significantly due to the non linear growth of calculation power when maintaining more ephemeral identities.

gwm
Mai 27, 2015

2 Methods and Material

As shown in Fig 1 we define the protocol on three different layers:

- A blending layer
  This is where messages are applied to the transport protocol. so far there are implementations for MIME based messages for SMTP and file transfer using XMPP messages.

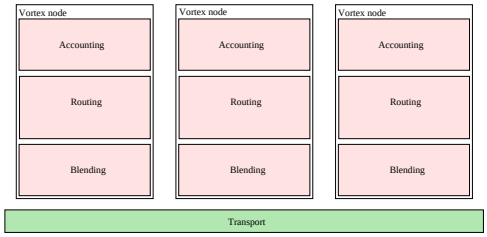

Fig. 1. Protocol stack.

- A routing layer
  This layer applies the logic to the message routing and prepares the message for the blending layer.
- An accounting layer
  This layer is a DoS and misuse protection. It keeps track of the transfers for each ephemeral identity and makes sure that queue and storage capacity are efficiently handled.

All three layers are connected through one or more common internet protocols such as SMTP or XMPP.

2.1 Blending Layer

This layer is a translation-only layer and blends traffic onto the transport protocol. Protocol features such as anonymity or redundancy do not rely on this level. This level should provide just enough. This blending must be done with great care to remain undetectable from a human context bound observer. If not done so problems as descrbed in [13] arise quickly. At the moment the system is limited to the two capabilities "embedd with offset" and "F5".

"Embedd with offset" is a plain embedding of a MessageVortex in a file attached to a message. The offset allows to issue first a header of some sort in order to improve blending (eg. for a PCM encoded WAV file). While this is considered a very weak protection, analysis to detect such a file on a global transport scale is very demanding due to the sheer mass to be analysed.

"F5" means applying the F5 algorithm to hide a VortexMessage within a random suitable JPEG image.

3 Results

The MessageVortex protocol is structured as follows.

4 Discussion

5 Conclusion

Acknowledgments

The authors would like to thank their families for being so patient with them, and may more **FIXME name them** for their thoughts on the paper.

References

[1] G. Danezis, R. Dingledine, and N. Mathewson, "Mixminion: Design of a type iii anonymous remailer protocol," in *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, May 2003, pp. 2–15. [Online]. Available: http://mixminion.net/minion-design.pdf

[2] C. Gülcü and G. Tsudik, "Mixing E-mail with Babel," in *Proceedings of the Network and Distributed Security Symposium - NDSS '96*. IEEE, Feb. 1996, pp. 2–16. [Online]. Available: http://citeseer.nj.nec.com/2254.html

[3] U. Möller, L. Cottrell, P. Palfrader, and L. Sassaman, "Mixmaster Protocol — Version 2," IETF Internet Draft, Jul. 2003. [Online]. Available: http://tools.ietf.org/pdf/draft-sassaman-mixmaster-03.pdf

[4] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The second-generation onion router," in *Proceedings of the 13th USENIX Security Symposium*, Aug. 2004. [Online]. Available: http://www.dtic.mil/cgi-bin/GetTRDoc?Location=U2&doc=GetTRDoc.pdf&AD=ADA465464

[5] R. Dingledine, M. J. Freedman, and D. Molnar, "The free haven project: Distributed anonymous storage service," in *Proceedings of Designing Privacy Enhancing Technologies: Workshop on Design Issues in Anonymity and Unobservability*, H. Federrath, Ed. Springer-Verlag, LNCS 2009, Jul. 2000. [Online]. Available: http://freehaven.net/doc/berk/freehaven-berk.ps

[6] S. Goel, M. Robson, M. Polte, and E. G. Sirer, "Herbivore: A scalable and efficient protocol for anonymous communication," Cornell University, Ithaca, NY, Tech. Rep. 2003-1890, Feb. 2003. [Online]. Available: http://www.cs.cornell.edu/People/egs/papers/herbivore-tr.pdf

[7] D. Chaum, "Untraceable electronic mail, return, addresses, and digital pseudonyms," *Communications of the ACM*, 1981. [Online]. Available: http://www.cs.utexas.edu/~shmat/courses/cs395t_fall04/chaum81.pdf

[8] ——, "The dining cryptographers problem: Unconditional sender and recipient untraceability," *Journal of Cryptology*, vol. 1, pp. 65–75, 1988. [Online]. Available: http://www.cs.ucsb.edu/~ravenben/classes/595n-s07/papers/dcnet-jcrypt88.pdf

[9] L. Øverlier and P. Syverson, "Locating hidden servers," in *Proceedings of the 2006 IEEE Symposium on Security and Privacy*. IEEE CS, May 2006. [Online]. Available: http://tor-svn.freehaven.net/anonbib/cache/hs-attack06.pdf

[10] M. V. Barbera, V. P. Kemerlis, V. Pappas, and A. Keromytis, "CellFlood: Attacking Tor onion routers on the cheap," in *Proceedings of ESORICS 2013*, Sep. 2013. [Online]. Available: http://www.cs.columbia.edu/~vpk/papers/cellflood.esorics13.pdf

[11] A. Biryukov, I. Pustogarov, and R. P. Weinmann, "TorScan: Tracing long-lived connections and differential scanning attacks," in *Proceedings of the European Symposium Research Computer Security - ESORICS'12*. Springer, Sep. 2012. [Online]. Available: http://freehaven.net/anonbib/papers/torscan-esorics2012.pdf

[12] A. Biryukov, I. Pustogarov, and R.-P. Weinmann, "Trawling for tor hidden services: Detection, measurement, deanonymization," in *Proceedings of the 2013 IEEE Symposium on Security and Privacy*, May 2013. [Online]. Available: http://www.ieee-security.org/TC/SP2013/papers/4977a080.pdf

[13] M. Abadi, M. Burrows, M. Manasse, and T. Wobber, "Moderately hard, memory-bound functions," *ACM Transactions on Internet Technology (TOIT)*, vol. 5, no. 2, pp. 299–327, 2005. [Online]. Available: http://www.isoc.org/isoc/conferences/ndss/03/proceedings/papers/2.pdf

Martin Gwerder Martin Gwerder was born 20. July 1972 in Glarus, Switzerland. He is currently a doctoral Student at the University of Basel. After having concluded his studies at the polytechnic at Brugg in 1997, he did a postgraduate study as a master of business and engineering. Following that, he changed to the university track doing an MSc in Informatics at FernUniversität in Hagen. While doing this he constantly broadened his horizon by working for industry, banking and government. His interests are in the field of networking related problems dealing with data protection, distribution, confidentiality and anonymity.