Lil'Sis' (LILS.IS)

Executive Summary

Peter Waher

ABC4.IO Chile SpA, Fransisco Soza Cousiño 610, #1601, Concon, V, Chile
peterwaher@hotmail.com

Abstract. LILS.IS, or Lil'Sis'™, as opposed to "Big Brother" is an information collaboration application and social networking tool protecting the privacy, information integrity and ownership of its users. It does this by making sure information in any accessible form does not leave the computers of the users sharing it (unless users explicitly choose to). It is never stored or processed on central processing nodes, except when they act as brokers of opaque end-to-end-encrypted blocks of information. Communication between peers is always End-to-End encrypted using state-of-the art cryptography, and Peer-to-Peer when possible.

Keywords: Decentralization, Information, Collaboration, Social Networking, Security, Privacy.

1 Introduction

Managing privacy, confidentiality, and security of information on the Internet, or across domains using the Internet as a communication medium, is a difficult problem. Traditional systems are built using a client/server architecture, where clients rely on servers to process information on the clients' behalf. If you don't need to communicate with others, you can work privately on your own computer, disconnected from the Internet. But as soon as you need to interoperate with others, inside your own organization, or between organizations (across domain boundaries), servers promising to relay your information to others also get access to your information. This leads to a lot of security and privacy problems as operators of this server infrastructure get access to your information. Backdoors, or inadvertent access through curiosity, or a profit motive behind using or disclosing your information, malpractice, corruption, industrial espionage, intelligence operations or malicious hacking of service providers, put your sensitive information at a definite risk. Any person, organization or company that works with private, sensitive, confidential, or classified information *should avoid using services based on a centralized client/server architecture* for these reasons.

On-site central processing

There exist many solutions for information collaboration today. These include e-mail, electronic bulletin boards, social networks, instant messaging applications, etc. Almost all of them rely on centralized processing on the Internet. Some companies realize their users are concerned with security and privacy and offer them to host on-site servers they can control instead. But this does not resolve the problem, for several reasons:

- IT personnel in the department get access to sensitive information, even if they should not.
- Support-personnel from the suppliers also get access to the information, when they operate, maintain or service their software or equipment.
- Localized servers may still execute code and report back to centralized services, sending information about usage, etc. This may include sensitive information.
- Malicious software in the network might also get access to these servers and extract information (so called lateral attacks).

Cross-domain interoperation

The problem of using on-site servers to host information collaboration services becomes even clearer when information exchange must occur across domains. If a company A hosts its own local server S_A for security reasons, what happens when it wants to collaborate with a company B? Should company B forfeit their security and privacy concerns, and use company A's server S_A? Or vice versa, should company A use company B's server S_B? Should information be shared between servers S_A and S_B, doubling the risks of data breaches? Or should a third organization C be created with a server of its own S_C, used for collaboration between companies A and B? What happens when projects require collaboration between more companies? Must each such constellation require its own server, multiplying risks for data breaches as complexity increases?

For practical reasons, you cannot combine privacy and security on one hand, and centralized processing, even at a domain level, on the other hand. This approach must be abandoned for a new approach. Servers must not be part of processing the information being shared between users. Instead, they must act as *brokers* only, relaying connections among peers, in a federated manner across domains, in a standardized way, for interoperability. Peers on the other hand, communicate with each other directly, exchanging the information being processed, in an end-to-end encrypted manner. Brokers also provide a means for peers to identify themselves globally (*digital identity*), in a secure manner, and peers are given the responsibility to authorize access to their information based on such identities of the other users.

Secure information collaboration applications are built on a *Peer/Broker* architecture, rather than a Client/Server architecture (or Master/Slave architecture).

Profit motive

The problem becomes even worse by the fact that many people have grown used to indiscriminately[1] using free software on the Internet, even for private, sensitive, and confidential information. This includes use of search engines, social network applications, messaging applications, e-mailing software, office applications, etc. Seldom do they consider the problem of where the companies creating the software get the revenue necessary to pay for developing, hosting, operating, and managing the software. If users do not pay for using the software, someone else pays for other motives. Often this motive, is to gain access to the information you provide using their services. They can use this information, either to profile you and sell generated meta-data about you to their customers, or they can use the information directly, for other purposes.

Do not indiscriminately use free software for information collaboration of private, sensitive, confidential, or classified information. A company is loyal only to its customers, who are the basis of their revenue, and therefore, their expenses and existence. Make sure you are that customer, and not somebody else (making you the product).

Ownership of information

Seldom is consideration given to who owns the information being processed by information collaboration tools. Many people do not realize that they give up any rights they might possess to the information they provide to free social networking services. This includes intellectual property rights, as well as copyrights. This would have been clear if they had read the license agreements before using the software.

If you own something, you must also show and demonstrate you own it, by protecting it. If you place a valuable item on the middle of the street and somebody takes it, you will not be able to demonstrate it is yours, and nobody will believe you, less reimburse you. The finder can keep it. The only way to demonstrate something is yours, is to protect it, preferably behind lock and key. An insurance company will not reimburse valuables taken from a house or a car unless they have been duly protected. The same holds true for information. *Information is only yours if you duly protect it*.

Value of information

Once a user controls the ownership of information, and can restrict access to it, it is also possible to assign a value to the information. If you provide the information for free to service providers, you lose this ability. They receive the information, and are also able to assign a value to it, without including you. By protecting the information, you retain the ability to assign a value to it. This means, you are also able to participate in any commercialization of the information. If a company can commercialize

[1] i.e., without thoroughly reading through, understanding and agreeing to all statements in the corresponding license agreements.

the information for you, you are at least able to get your corresponding share of the revenue.

Integrity of information

Recent years have seen a vast increase in attempts (many successful) to limit expression (threats), censor information (block access), expressly edit information without the consent of original authors (lost moral rights), or publish parts of it out of context (disinformation/misinformation), stifle reach of information (shadow-ban), etc. These are tactics typically used in information warfare and should not be a part of information exchange in a free, open, democratic, and civilized society. The reason such tactics are employed, is simply put *because they can*, and *they are cheap*.

Traditional system architecture makes such tactics very inexpensive, apart from being very effective tools for controlling opinion. All opinion operators need to do is to threaten service providers with reduced revenues, and they will comply. Just as public opinion can be stifled, by threats of personal cancellation, a service provider can modify its operation, by threats of revenue shortfalls, or other kinds of punishments.

The only way to protect the integrity of information and protect the right to freely publish or express it, is to make sure, technically, such threats cannot be employed. By making each user the publisher of their own information, you make sure there are no centralized agents that can be threatened, to stifle, block or alter the information and any speech. An oppressive authority would have to threaten every single user individually to achieve what today they only need to threaten one individual or small group of individuals. By protecting the information, making sure only the indented audience gets access to it, you also minimize the risk of adversaries gaining access to it, limiting their possibility to react maliciously to it.

2 Lil'Sis'™

Lil'Sis'™ is a social networking and information collaboration tool that presents an alternative to how information is usually managed, by providing solutions to the problems presented earlier. It provides a method that helps protect users' privacy on the Internet and provides a mechanism to define and control ownership of information.

Communication

The communication backbone used is based on the XMPP protocol[2]. As it is standardized by the Internet Engineering Task Force IETF[3], it provides an excellent means

[2] https://xmpp.org/

[3] https://datatracker.ietf.org/doc/html/rfc6120
https://datatracker.ietf.org/doc/html/rfc6121
https://datatracker.ietf.org/doc/html/rfc7622

to achieve real-time collaboration in a globally scalable manner across domains. It is used by many secure communication and instant-messaging applications (chat) that are globally recognized.

In its simplest form, it defines a Peer/Broker-architecture, allowing Peers to communicate with each other in a secure manner, regardless of network topology. It is secure, because, by default, others cannot connect to peers directly, instead all peers initially connect to brokers. The brokers collaborate using the standard, allowing peers to interconnect across domains. Once a peer is connected, brokers forward information addressed to it, and informs the recipient from where the information originated.

Once peers are connected, and can exchange public keys and connectivity options, peers can also choose to connect directly between each other, if possible. This allows them to establish secure peer-to-peer connections. Regardless of, if communication in Lil'Sis'™ is done peer-to-peer, or relayed over brokers, information is always end-to-end-encrypted first, to assure the integrity, privacy, and confidentiality of the information. The information is never processed (nor could it be) by any of the brokers involved in its readable form.

Information storage

Information you provide in Lil'Sis'™ is stored in encrypted local databases on the machines on which Lil'Sis'™ is installed. It is never stored on brokers, unless explicitly published by the user, for a broader audience. It is only shared with contacts (also running Lil'Sis'™) you have approved and given authorized access, in an end-to-end-encrypted manner, and peer-to-peer if possible. If brokers are used to relay the end-to-end-encrypted packet, they have no means to decrypt the packet. Only the contacts you have given access rights to specific information will be able to access it. When you update or remove the information from Lil'Sis'™, the change is propagated and will be updated or removed from your contacts as well.

Backups

As servers have no access to your information, and cannot make backups of it for your sake, Lil'Sis'™ handles backups for your automatically. Every day, month and year, an encrypted backup is generated of your information. These backups are stored for a given number of days, months, and years respectively. You can then choose how the backups are to be duplicated, for resilience. A network of trusted Lil'Sis'™ users can, for instance, store the backups of each other, to make sure a backup survives the failure of one or more of the machines involved. You can also choose to store the encrypted backups on servers in the network.

3 Features

Lil'Sis'™ supports many features users have come to expect for information collaboration:

Posting and timelines

- Posting information in timelines, syndication among contacts, replying to posts and replies, nested to any level.
- Posted content can be restricted to any number of groups.
- Unrestricted content can be viewed by your approved contacts.
- You can annotate your posts with any number of tags, organizing them in a decentralized tag cloud.

Publishing information

- You can choose to publish posted content, to your broker, making it accessible by anyone with access to the broker. If the broker is a TAG Neuron™, such published content can be made available on the web. This makes it possible to publish information to pages publicly on the web.
- You can subscribe to published content from any broker, making it easy to follow information flows from sources outside of your contact list.

Chat and Instant Messaging

- You can chat with individual contacts privately.
- You can engage in group-chats, using Multi-user Chat rooms hosted on any broker.

Content

- Information in posts, replies, messages, etc. can consist of formatted text, diagrams, images, video or audio, file attachments (file sharing) or combinations thereof, and more.
- Content is synchronized between peers, when they are simultaneously online, and the corresponding access rights have been authorized.
- You can synchronize folders with your contacts, either unidirectionally, or bidirectionally. This makes it possible to have shared working folders that are synchronized between members of groups.
- You are notified when new content is available, have been updated or removed, or replied to, etc.

Contacts

- You can have any number of contacts in your contact list ("roster")
- You can group your contacts into any number of groups.
- You can restrict content to specific contacts, or groups of contacts.
- Each participant can provide three levels of profile information: One publicly available, one available to those with access to your domain, and one available only to your approved contacts.

Operation

- Lil'Sis'™ is only online when the machine is online. Closing a PC, makes the user go offline from the network.
- You can create Lil'Sis'™ proxies, where you delegate the right to operate Lil'Sis'™ to designated contacts. These contacts can post from the Lil'Sis'™ proxy, from their own installations of Lil'Sis'™. (Source will still be visible to recipients.) This is a useful feature, when creating installations (for instance, for teams), that are not operated directly by humans, or give voice to a group of people, and that voice must be powered and online 24/7.
- Lil'Sis'™ provides a secure HTTPS-based API that makes it possible to create services that interact with it, always with the express approval of the user involved.
- Lil'Sis'™ is automatically updated with security updates and feature updates, as such become available, to minimize the risk of running obsolete software. Updates are published from the broker and can be delayed if security personnel require.
- Lil'Sis'™ can act as an application host to other services, such as ABC4.IO[4].

Transparency

- Lil'Sis'™ makes sure that each user can gain access to its own direct communication in real-time, to provide transparency, and demonstrate what is being communicated, and how it is being communicated. This is a technically advanced feature but allows people that are skilled in cybersecurity and communication technology to validate the claims made in this document.

[4] See ABC4.IO, Executive Summary, 2021-08-25.

Add-Ons

For security reasons, certain features are not available by default (and cannot be enabled through configuration, as the executable code is not included). When creating very secure software, that can have a *dual use*[5], it is very important to make sure certain technologies do not end up in the hands of belligerent or criminal organizations. Civilian users rarely require such features. They are therefore listed as add-ons, as they can only be provided to organizations having been approved for such use.

- *Deniability* is a specific feature of cryptographic algorithms, that make it possible to deny the origin of a message, if it has been leaked or breached. Typical civilian people and organizations have no need of this feature. Cryptographic algorithms in Lil'Sis'™ have an exceptionally high security strength. But if information is breached (for instance through access to an unlocked computer), and data is leaked, the damage has already been done. There is little point in being able to deny the origin (unless you are a whistle-blower or a criminal or terrorist), as such denial only have validity in courts of law. A whistle-blower organization can be seen as having a legitimate use of deniability, to protect whistle-blowers from recrimination from a much more powerful adversary or organization, while criminal and terrorist organizations do not.
- *Tracker of information-flow* is a feature, that permits Lil'Sis'™ to modify information in such a way that different contacts receive different versions of the information. If they in turn share the information, each of their contacts receiving the information gets a unique copy of it, and so on, when they share it. If the information leaks, it is possible to get an idea of the path the information has travelled in the network before it leaked, if access to the leaked information is obtained. As this information can pinpoint individuals involved in the leak, which can have serious consequences for the individual (depending on use case), such a feature is not available by default.

History

The idea behind Lil'Sis'™, was seeded a long time ago. As public social networks started to censor and control people, as well as using their information against them without their consent, the need for a Lil'Sis'™ alternative became acute. The name LILS.IS was registered 2015 and soon thereafter, development began. A presentation of Lil'Sis'™ was held at *Congreso Futuro* in 2017[6]. Due to legal complications, commercialization of Lil'Sis'™ was halted slightly thereafter, and could not be resumed until recently.

[5] https://ec.europa.eu/trade/import-and-export-rules/export-from-eu/dual-use-controls/
[6] https://www.lun.com/Pages/NewsDetail.aspx?dt=2017-01-11&PaginaId=4

Summary

Lil'Sis'™ is a secure information collaboration and social networking tool, that lets users collaborate in teams, within organizations, and across organizational boundaries (across domains). It is secure because information is stored and processed only on the machines of its users. Users have full control over who has access to what. Information is never shared with 3rd parties or processed on servers, unless explicitly stated by the user (for instance, when publishing information openly).

Lil'Sis'™ has a license fee, which finances the development, maintenance, and support of the technology. This fee makes it possible for Lil'Sis'™ to be loyal to its users, and not to other customers. There is no method for developers or operators of Lil'Sis'™ software and infrastructure, to get access to, and use personal and unpublished information shared by its users, unless the users share this information explicitly.