PRETTY EASY PRIVACY

www: pep.foundation
mail: sva@pep.foundation
IRC: #PrettyEasyPrivacy on Freenode
twitter: @pepFoundation @sva
hashtags: #PrettyEasyPrivacy #PrivacyByDefault

p≡p

Privacy by Default.

PGP / GPG

PGP ≡ Pretty Good Privacy

Created by Phil Zimmermann in 1991

see RFC 1991 (in 1996)

Zimmermann had been a long-time anti-nuclear activist, he created PGP that people might securely use BBSs and securely store messages and files

OpenPGP ≡ standard/specification

see RFC 2440 (in 1998) and 4880 (in 2007)

GPG ≡ GNU Privacy Guard

Created by Werner Koch 1999

from Free Software Foundation

(most common implementation of PGP)

What is Crypto?

mathematical way to make the data only readable for:

the one sending it

the one receiving it

→ end-to-end encryption

p≡p

Privacy by Default.

encrypt (or encode)

→ Code, Cipher,

or Key

p≡p

Privacy by Default.

Simple encoding/key:

Hello

olleH

p≡p

Privacy by Default.

Software helps:

Hello

hQEMAy4io41ThT7gAQgAqF7ljcgd

p≡p

Privacy by Default.

End-to-End

...only you and me have the key and no one else can listen or modify.

⋮

⋮

```
uSMWsh3zbWke8DUmY+Lf9Ssy2waJkE+gaJKhxplD6CWfL96vgXn3N/bBVg2+SCmt
UV/btwupjojluio1cLS0X85glj85sfeALHZGDzRTe7kuMXSqY9A+ZEpYIGybGkLk
8EjFZOqgDNRZRVe2mXpu7EOEwXEuIl2cANk5iXaVanAHGSMubUEzwkZWxvfHdPSZ
DWK9AYBRyIr62k8W7/rvpI8T8RtuinPbVWl5sLe7/x0smFvVfYj0Cy+UakOLgN08
4yghqyWWY7Hzc1Xq+UQrVib8CVnk5h/WQotuOshBmdLpAWMYkbNV3eJMxQ4xqx0u
```

p≡p

Privacy by Default.

Asymmetric Encrpytion

Also known as: “public-private-key-encryption”

Everyone has own key-pair:

Public Key:

Available for everyone!

Private Key:

Has to stay hidden!

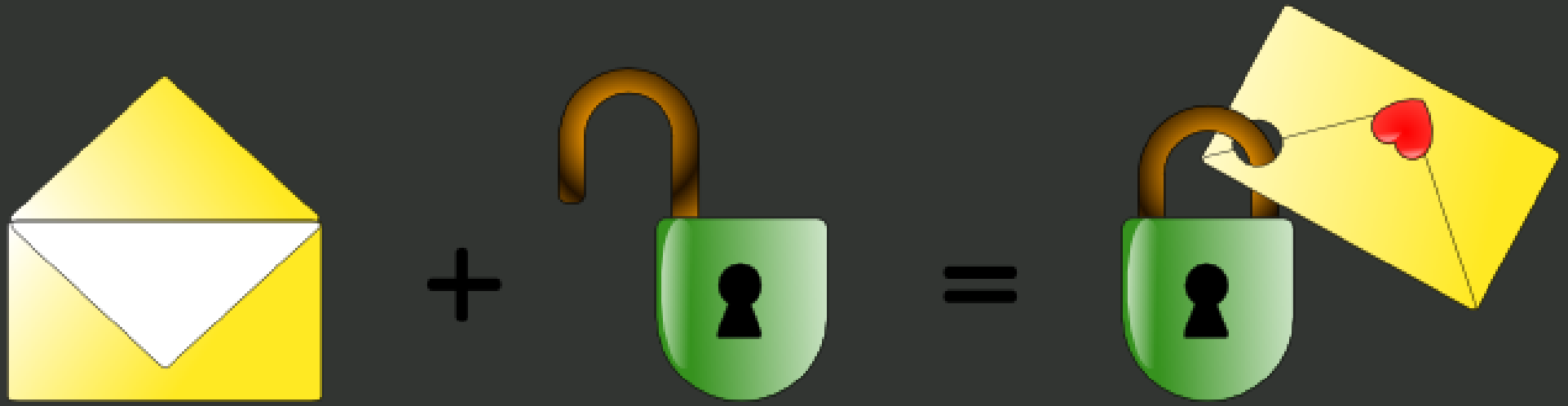
p≡p

Privacy by Default.

En-Crypt!

Bob uses the open lock / public key
from Alice to lock/encrypt the message.

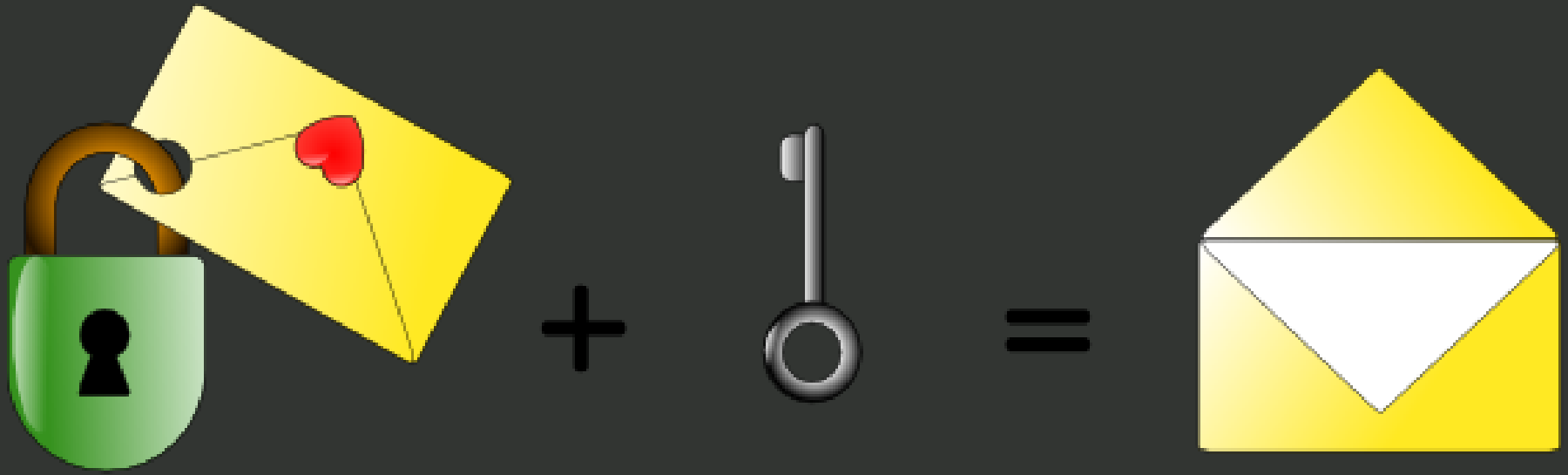
Once closed, he is not able to open it any more.



De-Crypt!

Alice uses her secret key to unlock/ decrypt the message from Bob.

Alice is **the only one** able to open this message.



4 rules-of-thumb

...when it's about encryption

If it's not open source, don't use it!

The older, the more likely it is secure.

Keep an eye on the news and recent happenings...

Where does it come from? Who is developing it?

p≡p

Privacy by Default.

4 rules-of-thumb
on GPG / PGP:

[x] free/libre open source software

[x] pretty old, in use ever since

[x] nothing in the news, still unbroken

[x] private persons and non-profit-foundations

p≡p

Privacy by Default.

Sequoia / A New OpenPGP

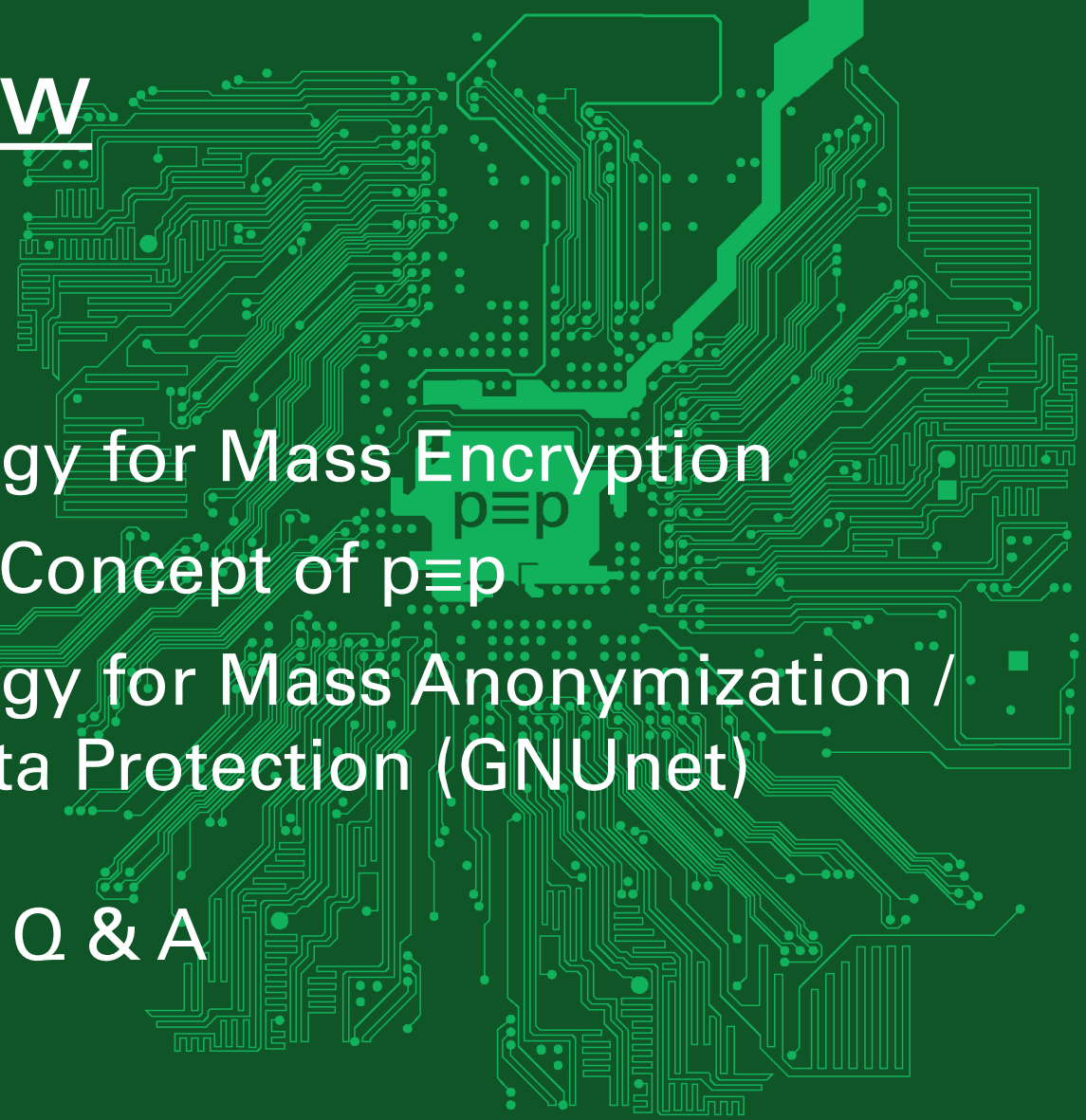
"...developers formerly working on GnuPG …
...create a new opinionated OpenPGP implementation."

A Cool OpenPGP Library
Does not dictate how OpenPGP is used
Keys are automatically kept up-to-date
Key rotations approximating forward secrecy

Written in Rust
Modern code base
Spatial and temporal memory safety: no leaks, no use-after-free, no out-of-bounds access, no race conditions
C API

p≡p
Privacy by Default.

Overview



0 – Intro

1 – Technology for Mass Encryption

2 – General Concept of p≡p

3 – Technology for Mass Anonymization / Meta Data Protection (GNUnet)

Summary + Q & A

p≡p

Privacy by Default.

0.1: What is p≡p?

...software for various platforms to easily use existing crypto tools (like GnuPG) ⇒ Pretty Easy

...designed to encrypt all digital written communication (with the starting point of email) ⇒ Privacy by Default.

...encrypts automatically with whatever (most privacy-enhancing) crypto standard available ⇒ Privacy by Default.

All end-user software must be hassle-free and zero-touch. ⇒ Pretty Easy!

p≡p

Privacy by Default.

0.2: What is p≡p not?

...not yet-another-crypto-tool with closed user base.

...not a (centralized) platform provider.

...not implementing any own crypto.

...not replacing any existing crypto tool per se.

… not just an email encryption tool:
that's just the beginning \o/

p≡p

Privacy by Default.

0.3: Who is p≡p?

We see ourselves as *cypherpunks.*

We want to roll out **mass encryption**
to optimize the costs of mass surveillance!

We want to make the use of crypto pretty easy:

The **developer plugs it** into apps.
The **user just uses it.**

By default.

p≡p
Privacy by Default.

0.4: Who? Cypherpunks?

...are actively engaged in **making the networks safer**

...advocate widespread use of strong cryptography […]
as a **route to social and political change**.

...aim to achieve privacy and security through
proactive use of cryptography.

Cypherpunk manifesto: 9th March 1993, Eric Hughes

p≡p
Privacy by Default.

0.4: Cypherpunk Manifesto

"Privacy is necessary for an open society in the electronic age.

Privacy is not secrecy.

A **private** matter is something one **doesn't want the whole world to know**, but a **secret** matter is something one **doesn't want anybody to know**.

Privacy is the power to selectively reveal oneself to the world."

9th March 1993, Eric Hughes

p≡p

Privacy by Default.

0.4: Cypherpunk Manifesto

"We cannot expect governments, corporations, or other large, faceless organizations to grant us privacy (…)

We must **defend our own privacy** if we **expect to have any**.

(...) We know that **someone has to write software** to defend privacy, and ... **we're going to write it.**"

9th March 1993, Eric Hughes

p≡p

Privacy by Default.

0.5: Who is sva?

Anthropology

Computer Science

Carpenter

cryptoparty.in
hillhacks.in
hackbeach.in
events.ccc.de

sva = unique address
in Internet and Web

(founded 1981)
Chaos Computer Club
Hackers without Borders
(founded 2014)

p≡p
Privacy by Default.

0.6: Declaration of Human Rights

Article 12

"No one shall be subjected to **arbitrary interference with his privacy, family, home or correspondence**, nor to attacks upon his honour and reputation.

Everyone has the **right to the protection of the law** against such interference or attacks."

http://www.un.org/en/universal-declaration-human-rights/

0.7: Quote

“I don't want to live in a world where everything that I say, everything I do, everyone I talk to, every expression of creativity or love or friendship is recorded.”

p≡p

Privacy by Default.

0.7: Quote

“I don't want to live in a world where everything that I say, everything I do, everyone I talk to, every expression of creativity or love or friendship is recorded.”

(Edward Snowden)

(TS//SI//NF) PRISM Collection Details

Current Providers

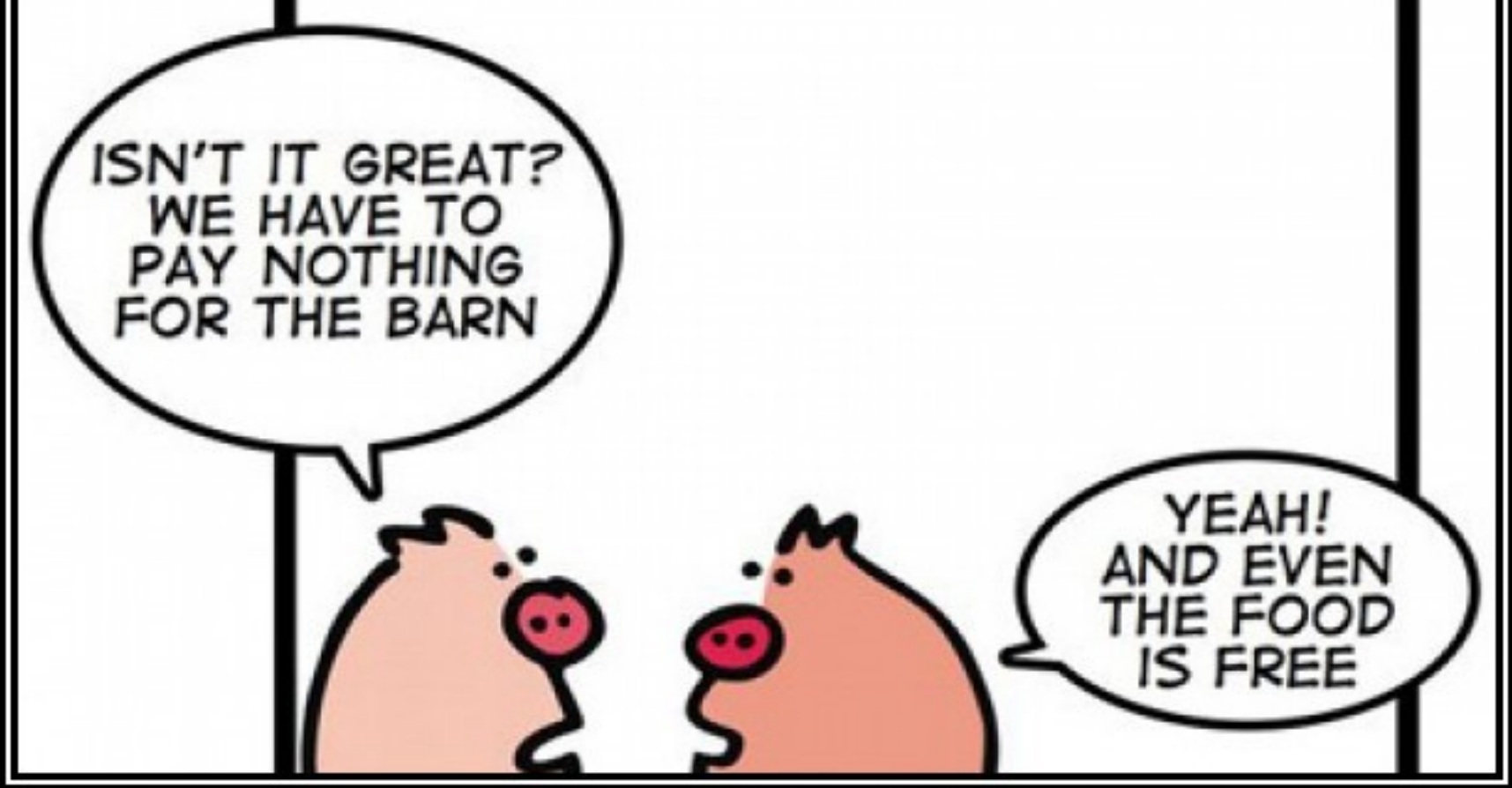
- Microsoft (Hotmail, etc.)
- Google
- Yahoo!
- Facebook
- PalTalk
- YouTube
- Skype
- AOL
- Apple

What Will You Receive in Collection (Surveillance and Stored Comms)? It varies by provider. In general:

- E-mail
- Chat – video, voice
- Videos
- Photos
- Stored data
- VoIP
- File transfers
- Video Conferencing
- Notifications of target activity – logins, etc.
- Online Social Networking details
- **Special Requests**

Complete list and details on PRISM web page:
Go PRISMFAA

p≡p
Privacy by Default.



FACEBOOK AND YOU

If you're not paying for it, you're not the customer. You're the product being sold.

0.8: Problem

Problem:

Online communication is visible like a postcard &
this world has mass surveillance

p≡p

Privacy by Default.

0.8: Problem & Solution

Problem:

Online communication is visible like a postcard &
this world has mass surveillance

Solution:

Encryption
Anonymization

p≡p
Privacy by Default.

0.8: Problem & Solution?

Problem:

Online communication is visible like a postcard &
this world has mass surveillance

Solution:

Encryption
Anonymization

Is it really a solution?

???

Encryption &
Anonymization
are there!

Since decades….

Still too hard to use!

p≡p
Privacy by Default.

0.8: Problem & Solution!

Problem:

Online communication is visible like a postcard &
this world has mass surveillance

Solution:

Mass Encryption == Privacy by Default.
Mass Anonymization == Privacy by Design.

p≡p
Privacy by Default.

0.9: pretty Easy privacy

p≡p does what the user *would want to* do

Instead of writing how-to guides
we write user expectations
into software and protocols,

to automatize all steps a user would need to carry out.

⇒ Taking away “crypto needs” from users view (like *https)*

p≡p

Privacy by Default.

Real-Time
General
Internet
Ops & Mgmt
Routing
Security
Transport
IRTF

New work
Chartering groups
BOFs

Other groups
Concluded groups
Non-WG lists

Documents
Search
Draft submission
Sign in to track docs

RFC streams
IAB
IRTF
ISE

Meetings
Agenda
Materials
Floor Plan
Past proceedings
Upcoming
Past
Request a session
Session requests

Other
IPR disclosures

Document	**Type**	Active Internet-Draft (individual)
	Last updated	2017-06-28
	Stream	(None)
	Intended RFC status	(None)
	Formats	plain text xml pdf html bibtex
Stream	**Stream state**	(No stream defined)
	Consensus Boilerplate	Unknown
	RFC Editor Note	(None)
IESG	**IESG state**	I-D Exists
	Telechat date	
	Responsible AD	(None)
	Send notices to	(None)

Email authors | IPR | ←References | →Referenced by | !Nits | Search lists

```
Network Working Group                                            V. Birk
Internet-Draft                                                 H. Marques
Intended status: Standards Track                                 Shelburn
Expires: December 31, 2017                                   pEp Foundation
                                                                 S. Koechli
                                                               pEp Security
                                                              June 29, 2017

             pretty Easy privacy (pEp): Privacy by Default
                          draft-birk-pep-00

Abstract

   Building on already available security formats and message transports
   (like PGP/MIME for email), pretty Easy privacy (pEp) describes
   protocols to automatize operations (key management, key discovery,
   private key handling including peer-to-peer synchronization of
   private keys and other user data across devices) that have been seen
   to be barriers to deployment of end-to-end secure interpersonal
   messaging.  pEp also introduces "Trustwords" (instead of
```

0.10: RFC / Internet-Draft

We started an Internet-Draft together with the ISOC-CH on the general pEp principles.

It's online and ready for discussion:

https://datatracker.ietf.org/doc/draft-birk-pep/

What does pEp do?

pretty Easy privacy (pEp): Privacy by Default
draft-birk-pep-00

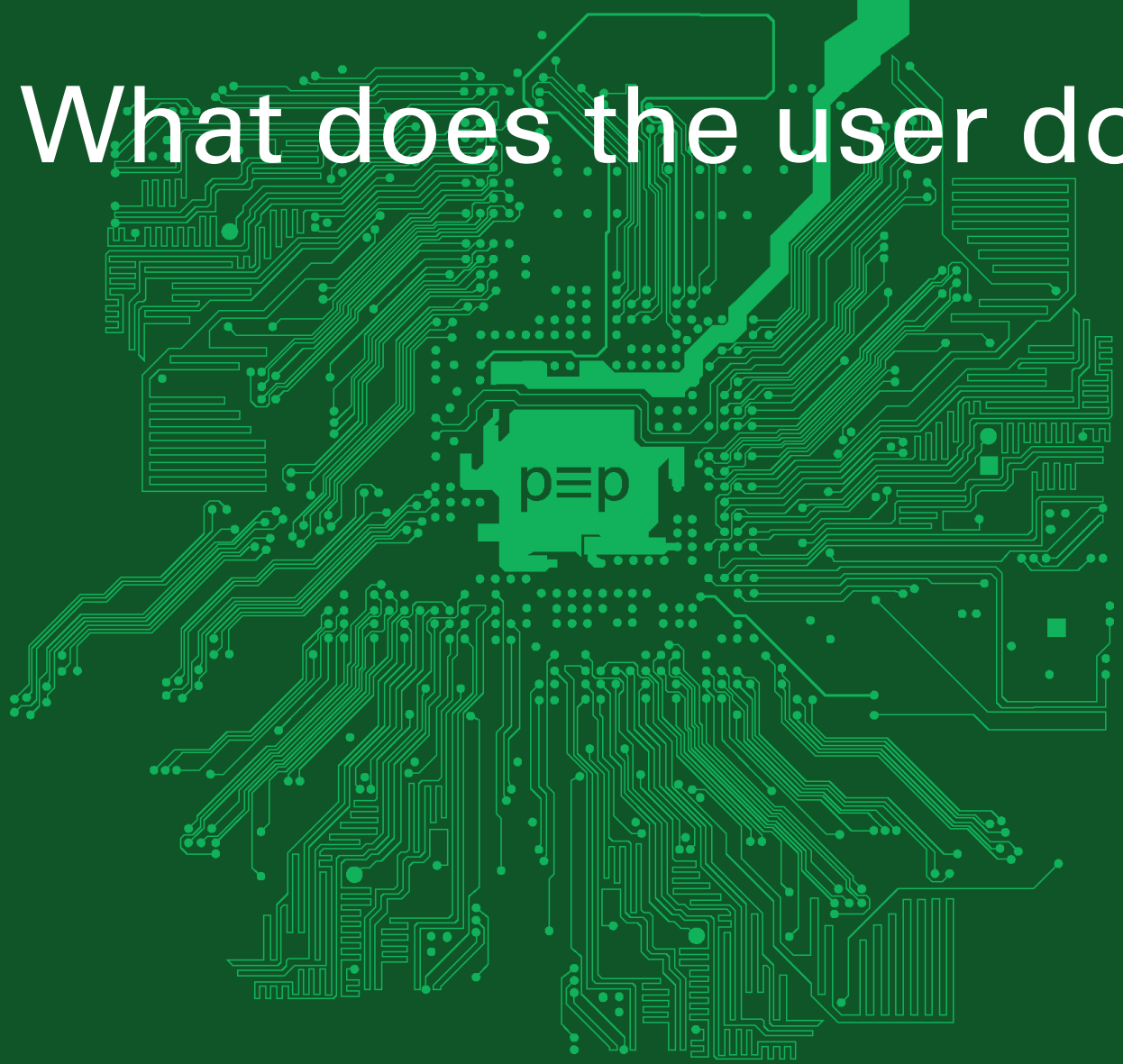
Abstract

Building on already available security formats and message transports (like PGP/MIME for email), pretty Easy privacy (pEp) describes protocols to automatize operations (key management, key discovery, private key handling including peer-to-peer synchronization of private keys and other user data across devices) that have been seen to be barriers to deployment of end-to-end secure interpersonal messaging. pEp also introduces "Trustwords" (instead of fingerprints) to verify communication peers and proposes a trust rating system to denote secure types of communications and signal the privacy level available on a per-user and per-message level. In this document, the general design choices and principles of pEp are outlined.

https://datatracker.ietf.org/doc/draft-birk-pep/

p≡p
Privacy by Default.

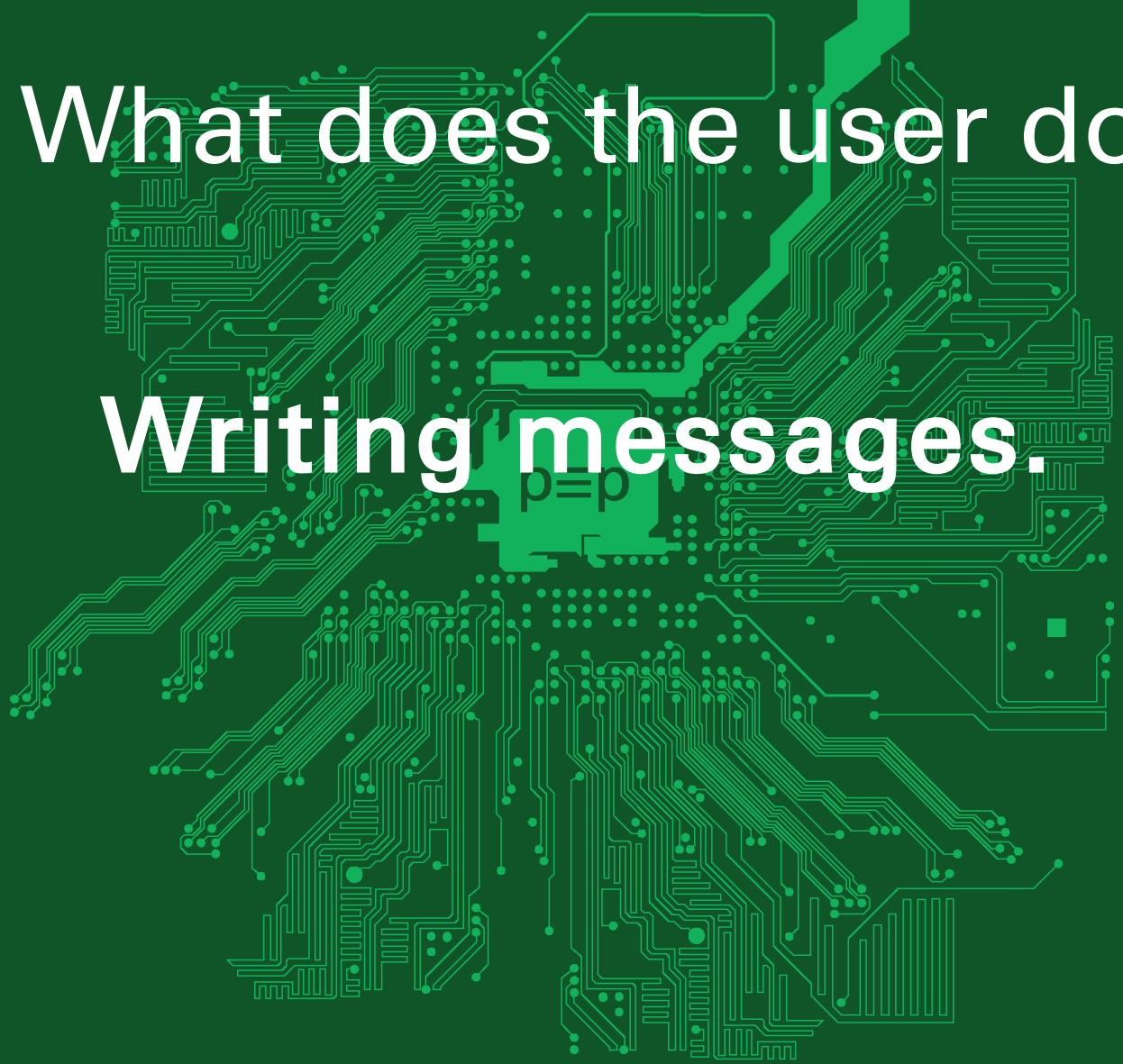
What does the user do?



p≡p

Privacy by Default.

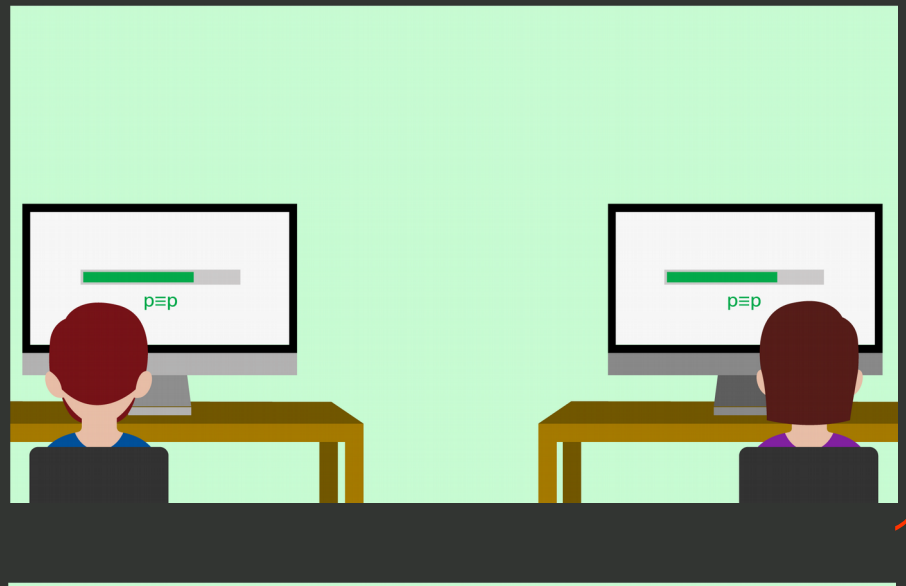
What does the user do?

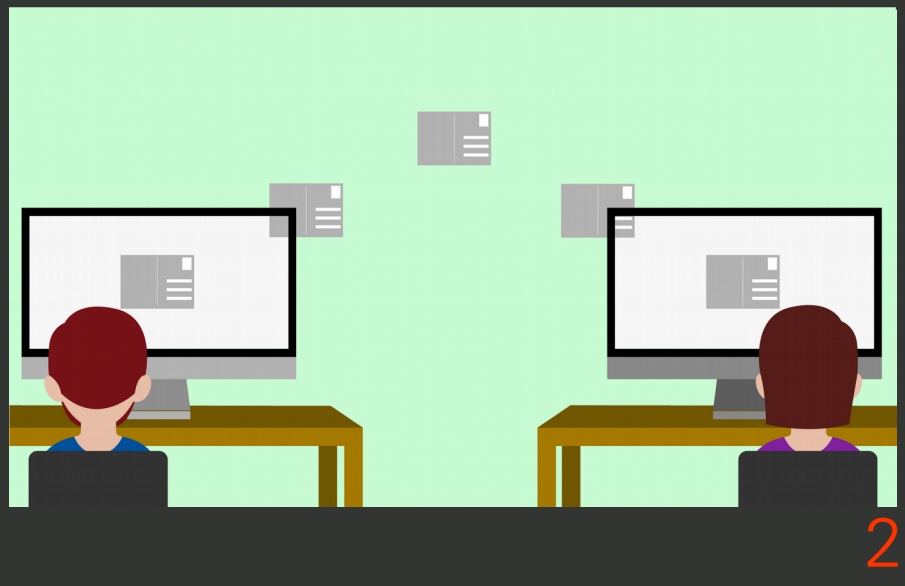


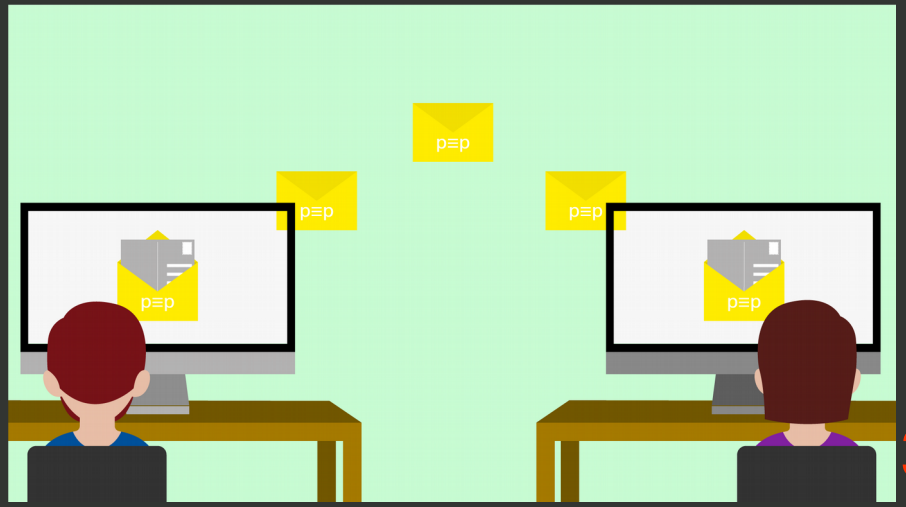
Writing messages.

p≡p

Privacy by Default.

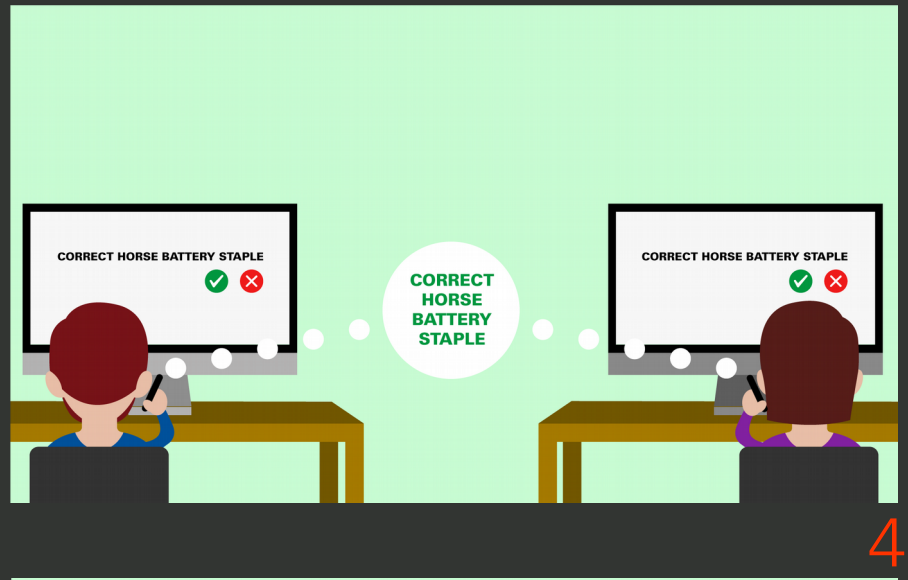




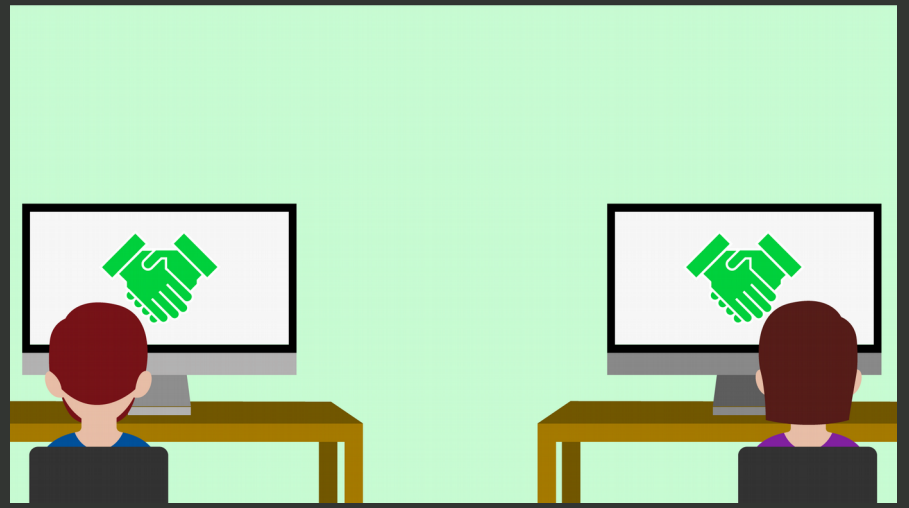


p≡p

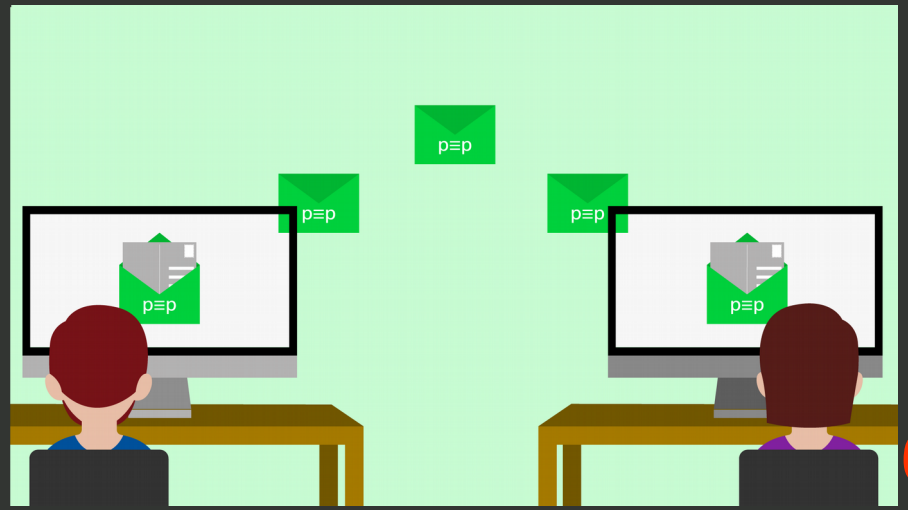
Privacy by Default.



4



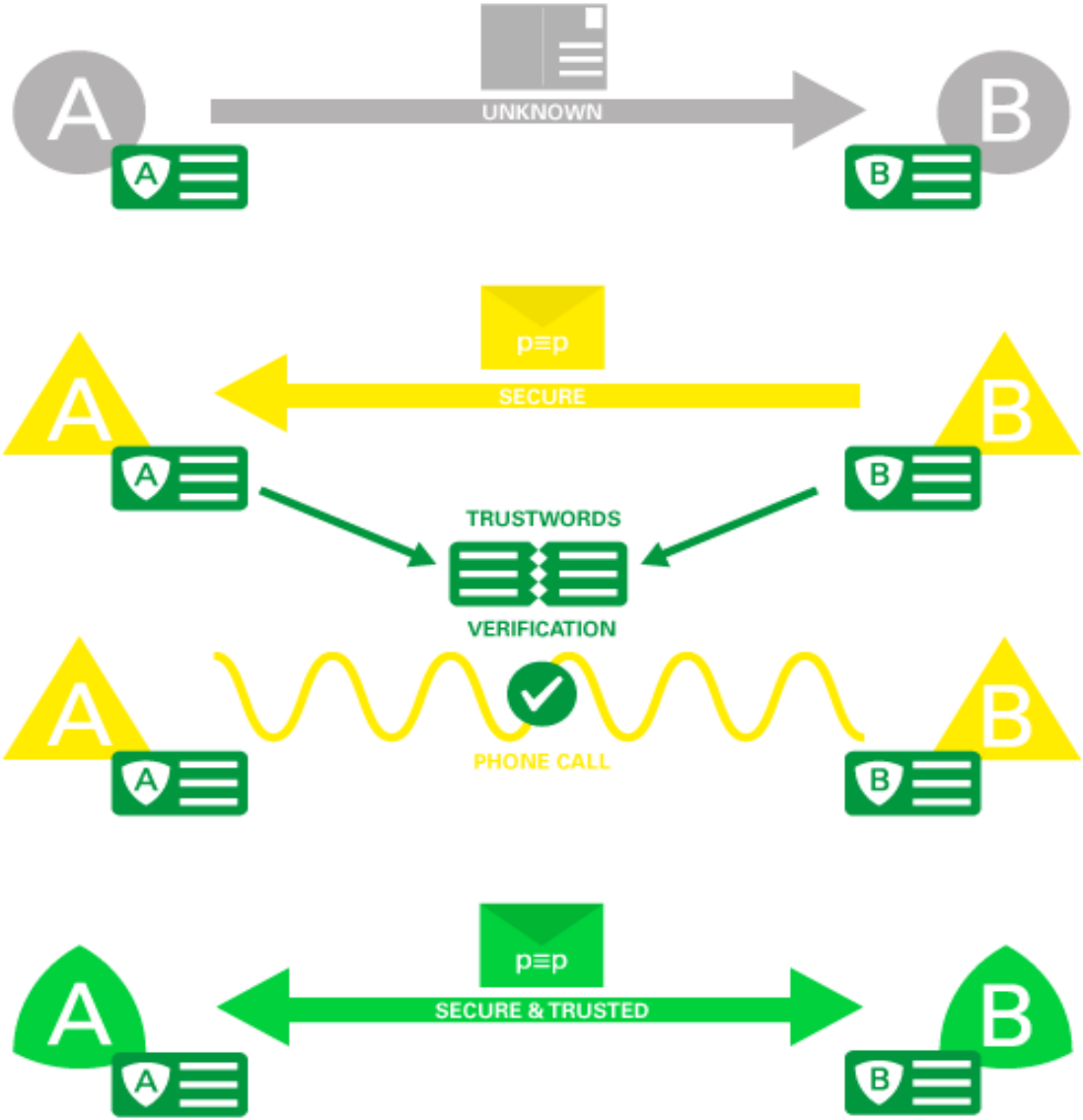
5



6

p≡p

Privacy by Default.



p≡p

Privacy by Default.

1: Technology: Overview

1.0. Architecture

1.1. Engine

1.2. Adapter

1.3. Applications

1.4. Organizational Forms

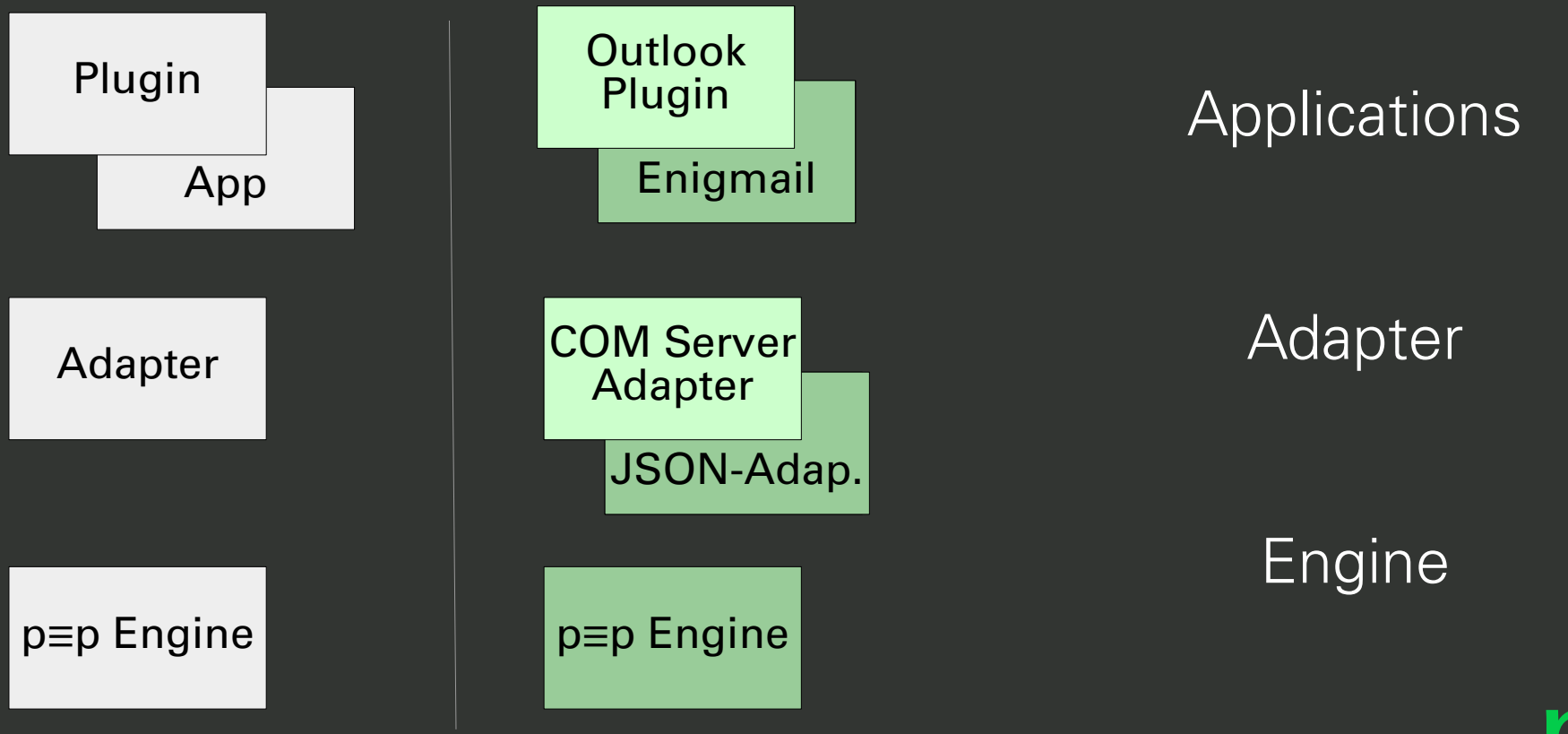
1.5. Repositories

1.6. Developing Platforms

p≡p

Privacy by Default.

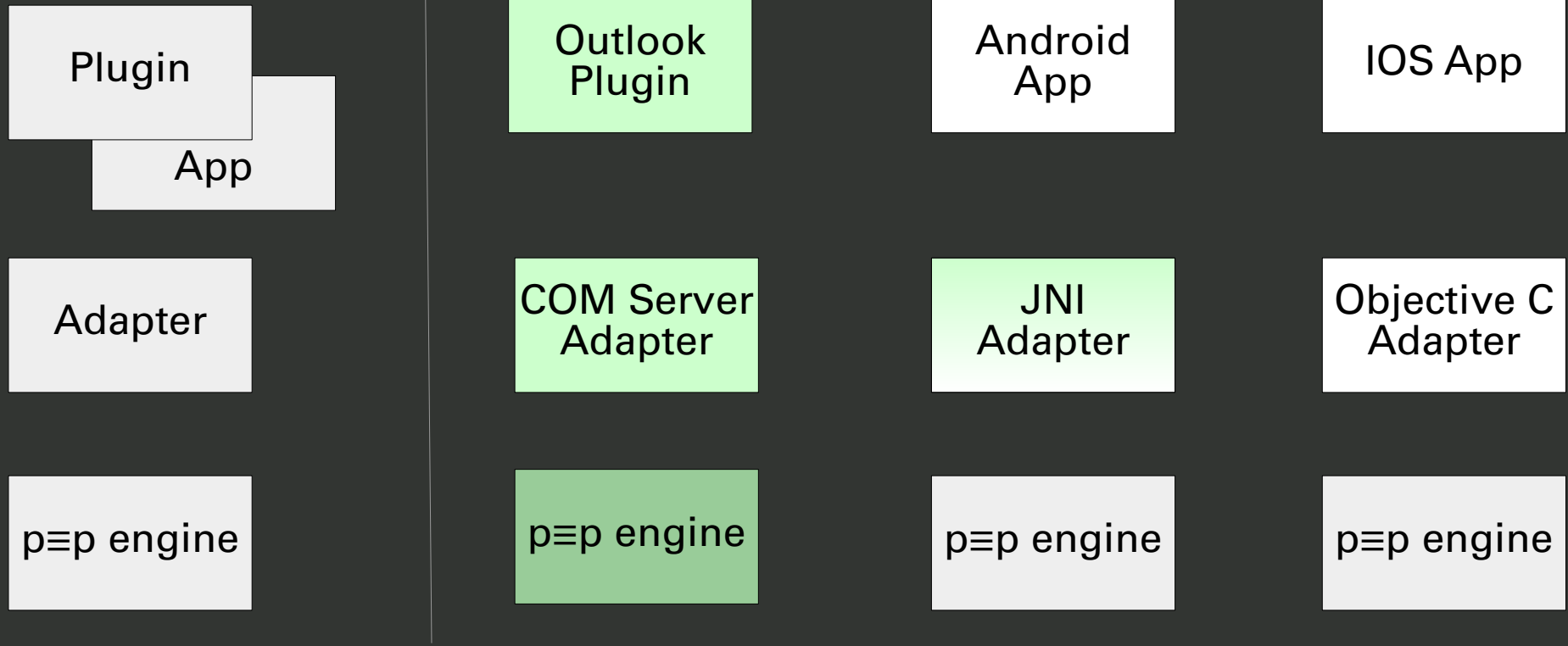
1.0. p≡p Tech: Architecture



p≡p

Privacy by Default.

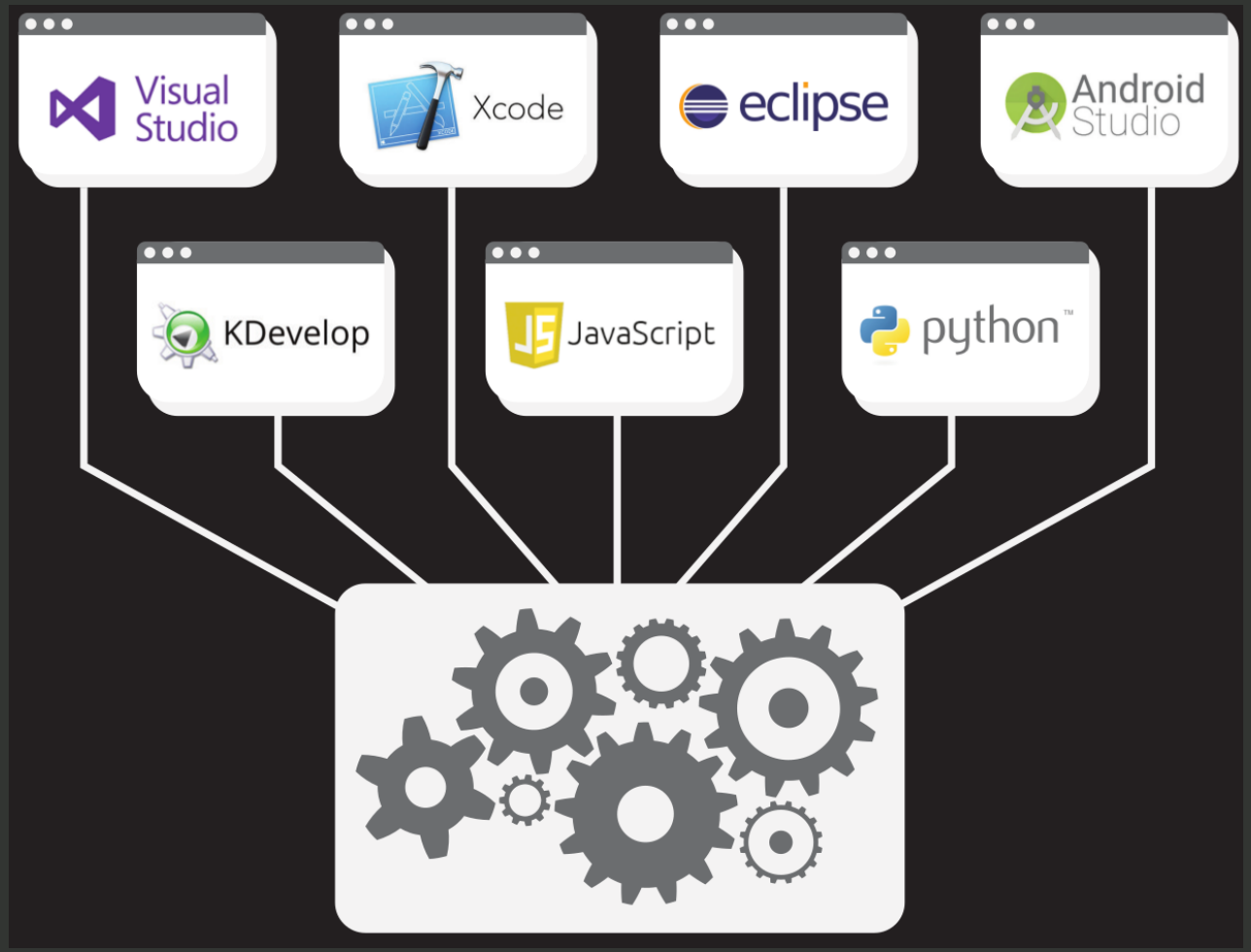
1.0. p≡p Tech: Architecture



p≡p

Privacy by Default.

1.0. p≡p Tech: Architecture



p≡p

Privacy by Default.

1.1. p≡p Tech: Engine

Engine drives several crypto standards on different digital channels / message transport protocols.

Written in C99,
~10.000 lines of code,
with regular code audits.

p≡p
Privacy by Default.

1.1. p≡p Tech: Engine

Engine drives several crypto standards on different digital channels / message transport protocols.

Written in C99,
~10.000 lines of code,
with regular code audits.

Not meant to be used in application code directly!!1!!11!!

p≡p
Privacy by Default.

1.1. p≡p Tech: Engine

As developer you can just plug’n’play the engine,
which means you don’t have maintain any crypto.

p≡p

Privacy by Default.

1.1. p≡p Tech: Engine

As developer you can just plug’n’play the engine,
which means you don’t have maintain any crypto.

Ehm, wait, what?

p≡p
Privacy by Default.

1.1. p≡p Tech: Engine

As developer you can just plug'n'play the engine,
which means you don't have maintain any crypto.

Ehm, wait, what?

Code Audits: https://pep.foundation/docs/code-audits/

p≡p
Privacy by Default.

1.1. p≡p Tech: Engine

Takes care of:

Messaging functions

Cryptotech services (crypto API)

Fully automated key management services

Trust rating

Knows about:

Transport Protocols / Message Transports

In the future:

Meta Data Protection via GNUnet (Anonymization)

p≡p

Privacy by Default.

1.1. p≡p Tech: Engine

Does:

- Decryption & Encryption

- MIME encoding & decoding

- Message processing for the adapter

- Key management: generation, verification, blacklisting

- Key synchronization of same account between devices

p≡p

Privacy by Default.

1.2. p≡p Tech: Adapter: What?

… is a language/environment-specific interface between the engine API and an application development environment (like a programming language or IDE).

Basically adapters serve bindings.

Adapter	Example Languages
COM Server Adapter	C#, C++, VB.Net
JNI Adapter	Java (e.g. Android)
JSON Adapter	Javascript
ObjC Adapter	Swift (iOS, macOS)
Python Adapter	Python
C++/Qt Adapter	C++, Qt

p≡p

Privacy by Default.

1.2. p≡p Tech: Adapter: How?

(1) App makes calls to the adapter for the function it wants:

encrypt / decrypt / mime-encoding
getting trustwords / verifying identity
decide on trustlevel for identity and/or a particular message

(2) Adapter converts that into:
normalized, standardized form (for the engine)
for messages and makes the C library call.

~~~ engine magic ~~~

(3) Adapter gives the result back to the application

p≡p
Privacy by Default.
~~~

1.3. p≡p Tech: Apps: Current Implement.

Handles OpenPGP without hassle for the user:

Automatically encrypts
Encrypts the subject inline
Automatic key management
Import of existing keys
No keyserver or any other centralized infrastructure

Fingerprints ≡ Trustwords
Opt-in passphrase for keys
Disclaimer-Function
"Force-Protection"
"Passive-Mode"
Header encrypted & obfuscated
p≡pSync

p≡p
Privacy by Default.

1.3. p≡p Tech: Applications

MS Outlook via Add-in

Thunderbird via Enigmail/p≡p (Win, Linux, MacOS)

p≡p for Android (K-9-Fork)

p≡p for iOS (new MUA, Alpha)

… more to come … we've just started …
... KMail ... Mutt … browser plugins …
… your mail client here …
… SMS … Jabber … facebook/twitter/whatever …
… everything … you know … MASS-ENCRYPTION !!!1!! …

p≡p
Privacy by Default.

1.4. p≡p Tech: Organizational Forms

https://pep.foundation:

(Foundation)

Supporting Free Sofware;
Code belongs to the foundation

https://pep.security:

(Company)

Selling Applications/Plugins and Services

https://pep.coop:

(Cooperative)

Bringing people together (Memberships),
Cooperations with other projects, Webplugins.

p≡p
Privacy by Default.

1.4. p≡p Tech: Other Websites (in progress!)

https://pep.software:

Download of Software (binary & code, *in progress*)

https://pep.community:

Forum, Mailing lists, Chat, etc. *(in progress)*

https://pep.news:

(News of the pEp-Universe, not there yet...)

p≡p

Privacy by Default.

1.5. p≡p Tech: Repositories:

Android: https://**pep-security.lu/gitlab**/android/pep/

Outlook: https://**pep-security.lu**/dev/repos/pEp_for_Outlook/

iOS: https://**pep-security.ch**/dev/repos/pEp_for_iOS/

Enigmail: https://**sourceforge.net**/p/enigmail/source/ci/master/tree/

Engine & Adapter & MISC: https://pep.foundation/dev/

Everything: https://pep.foundation/pep-software

https://pep.software

p≡p

Privacy by Default.

contact@pep.foundation

p≡p foundation.

HOME | ABOUT | P≡P SOFTWARE | SUPPORT P≡P | CONTACT

P≡P SOFTWARE

Home / p≡p software

p≡p foundation's approach

The foundation itself does not create end-user apps, add-ons and business-oriented solutions, but it provides the necessary core technology (p≡p engine and adapters) and multi-licensing strategy for doing so to a wide range of platforms. It otherwise supports Free/Libre and Open Source Software (FLOSS) to integrate p≡p into their projects.

That's the way the p≡p foundation takes to ensure that every person on every platform can finally have Privacy by Default, by having all their communications automatically encrypted and making the clear text message the exception.

Software ready to use

Released end-user software (for Android, Outlook and Thunderbird so far) can be obtained from **pep.software**.

For user documentation on p≡p for Android and Outlook, check the **p≡p security documentation**.

This Android, iOS and Outlook software is created by p≡p security, a commercial entity (incorporated in Switzerland and Luxembourg), implementing and shipping p≡p end-user software including professional support. Enigmail is a community project supported by all p≡p entities, including the foundation.

For more information about p≡p security, visit **pep.security**.

Upcoming end-user software

Very active work is being carried out to bring p≡p to the iOS platform; further desktop platforms (e.g., MS Outlook for macOS and Apple Mail) will be served as well as browser add-ons are to be shipped to encrypt web mail and other messaging.

Source code of p≡p's core

- **p≡p engine**: core to automatize crypto and key management
- **p≡p JSON Server Adapter**: JavaScript/JSON abstraction to p≡p (use-case: Enigmail)
- **p≡p COM Server Adapter**: C#/COM abstraction to p≡p (use-case: Outlook)
- **p≡p JNI Adapter**: Java Native Interface to p≡p (use-case: Android)
- **p≡p Objective-C Adapter**: Objective-C/Swift abstraction to p≡p (use-case: iOS)
- **p≡p Qt abstraction to p≡p**: Qt toolkit abstraction to p≡p
- **p≡p Python Adapter**: Python3 abstraction to p≡p
- All core code, including dependencies (YML2, NetPGP etc.): **here**.

Source code of end-user software

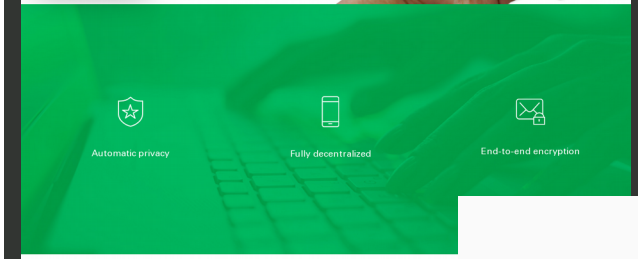
The foundation gave a trademark license to the commercial p≡p security entity, which is independent from the foundation and creates apps and add-ons for mass end-user and business markets. That entity primarily runs a business model similar to Red Hat's. All that end-user software is accompanied by the source code (in GNU General Public License v3), too:

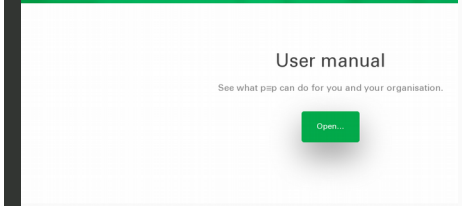
p≡p security projects:

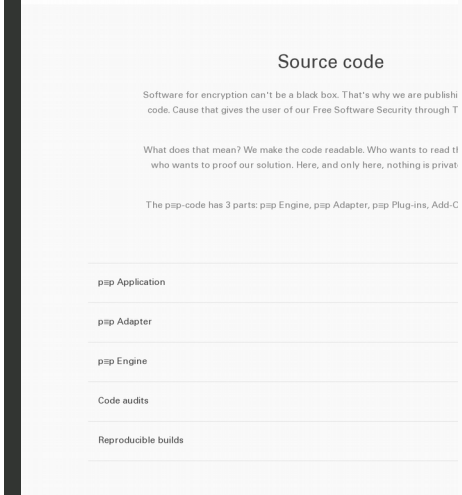
- **p≡p for Android** (on K9-fork app basis) **source code**
- **p≡p for iOS** (as a new Mail User Agent) **source code**
- **p≡p for Outlook** (as add-on) **source code**

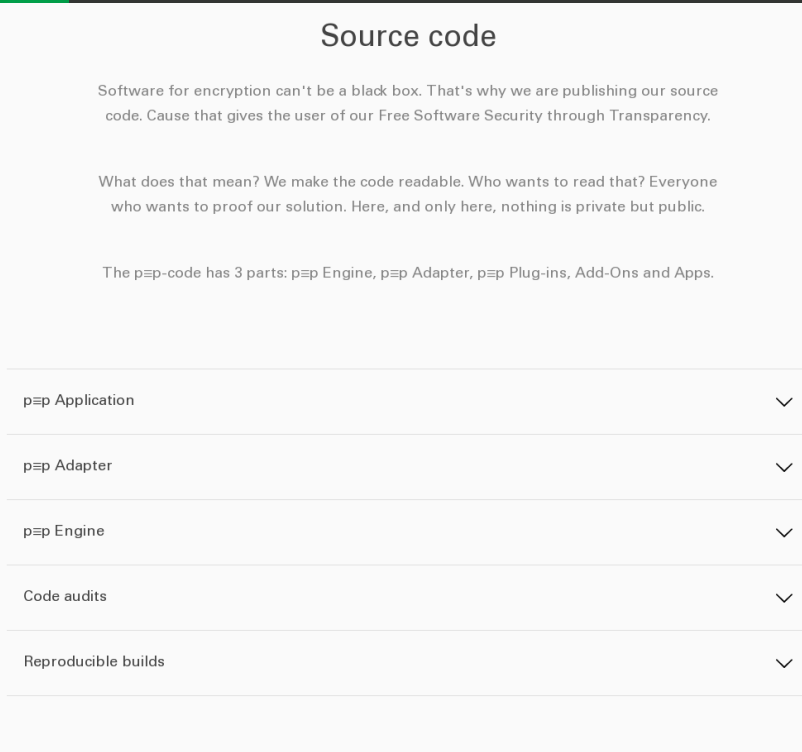
Community projects:

- **Enigmail/p≡p** **source code**









1.5. p≡p Tech: Repos of Foundation:

Repositories list

Name	Description	Contact	Last modified					
MessageModel	Modelling Message and Folder	p≡p development team <dev@pep-project.org>	Wed, 03 Oct 2018 15:17:44 +0200	zip	gz	bz2	RSS	Atom
downloadclient	client implementation for p≡p update server	p≡p development team <dev@pep-project.org>	Wed, 26 Apr 2017 13:19:32 +0200	zip	gz	bz2	RSS	Atom
enigmailpEp	Misc code for Enigmail/p≡p (cf. https://xkcd.com/1077/)	p≡p development team <dev@pep-project.org>	Mon, 23 Jul 2018 18:07:13 +0200	zip	gz	bz2	RSS	Atom
internet-drafts	p≡p I-Ds (IETF Internet-Drafts)	p≡p development team <dev@pep-project.org>	Fri, 14 Sep 2018 20:15:03 +0200	zip	gz	bz2	RSS	Atom
libAccountSettings	Platform-independent connection and server settings configuration	p≡p development team <dev@pep-project.org>	Wed, 08 Aug 2018 10:44:31 +0200	zip	gz	bz2	RSS	Atom
libpEpAdapter	C++ library for common structures used in p≡p adapters	p≡p development team <dev@pep-project.org>	Mon, 15 Oct 2018 21:34:37 +0200	zip	gz	bz2	RSS	Atom
netpgp-et	fork of netpgp (iOS adaptions and fixes)	p≡p development team <dev@pep-project.org>	Wed, 19 Sep 2018 12:22:17 +0200	zip	gz	bz2	RSS	Atom
pEpCOMServerAdapter	p≡p COM server adapter	p≡p development team <dev@pep-project.org>	Tue, 16 Oct 2018 01:24:49 +0200	zip	gz	bz2	RSS	Atom
pEpEngine	p≡p engine	p≡p development team <dev@pep-project.org>	Tue, 16 Oct 2018 12:30:03 +0200	zip	gz	bz2	RSS	Atom
pEpJNIAdapter	p≡p JNI adapter	p≡p development team <dev@pep-project.org>	Thu, 20 Sep 2018 11:17:06 +0200	zip	gz	bz2	RSS	Atom
pEpJSONServerAdapter	p≡p JSON adapter	p≡p development team <dev@pep-project.org>	Mon, 15 Oct 2018 16:48:27 +0200	zip	gz	bz2	RSS	Atom
pEpMIME	p≡p MIME library	p≡p development team <dev@pep-project.org>	Mon, 24 Apr 2017 15:06:52 +0200	zip	gz	bz2	RSS	Atom
pEpObjCAdapter	p≡p Objective-C (and Swift) adapter	p≡p development team <dev@pep-project.org>	Fri, 05 Oct 2018 16:36:10 +0200	zip	gz	bz2	RSS	Atom
pEpPythonAdapter	p≡p Python adapter	p≡p development team <dev@pep-project.org>	Thu, 04 Oct 2018 10:35:31 +0200	zip	gz	bz2	RSS	Atom
pEpQtAdapter	p≡p Qt adapter	p≡p development team <dev@pep-project.org>	Sun, 01 Oct 2017 23:50:53 +0200	zip	gz	bz2	RSS	Atom
pantomime-iOS	fork of pantomime (iOS adaptions)	p≡p development team <dev@pep-project.org>	Mon, 01 Oct 2018 12:01:07 +0200	zip	gz	bz2	RSS	Atom
yml2	>b's YML 2	Volker Birk <vb@dingens.org>	Sat, 08 Sep 2018 14:37:46 +0200	zip	gz	bz2	RSS	Atom

1.5. p≡p Tech: Repos

https://pep.foundation/dev

MessageModel	Modelling Message and Folder
downloadclient	client implementation for p≡p update server
enigmailpEp	Misc code for Enigmail/p≡p (cf. https://xkcd.com/1077/)
internet-drafts	p≡p I-Ds (IETF Internet-Drafts)
libAccountSettings	Platform-independent connection and server settings config
netpgp-et	fork of netpgp (iOS adaptions and fixes)
pEpCOMServerAdapter	p≡p COM server adapter
pEpEngine	p≡p engine
pEpJNIAdapter	p≡p JNI adapter
pEpJSONServerAdapter	p≡p JSON adapter
pEpMIME	p≡p MIME library
pEpObjCAdapter	p≡p Objective-C (and Swift) adapter
pEpPythonAdapter	p≡p Python adapter
pEpQtAdapter	p≡p Qt adapter
packages	Documentation for packages with p≡p software
pantomime-iOS	fork of pantomime (iOS adaptions)
yml2	>b's YML 2

p≡p

Privacy by Default.

1.6. p≡p Tech: Developing Platforms

iOS

Android

Linux

BSD

MacOS

Windows

p≡p

Privacy by Default.

2 – Concept: Overview

2.0. Privacy by Default

2.1. pretty Easy privacy

2.2. Peer-to-Peer and End-to-End

2.3. Free Software

2.4. Compatibility (Crypto & Transports)

2.5. Meta Data Protection

2.6. Summary

p≡p

Privacy by Default.

2.0. p≡p Concept: Privacy by Default.

p≡p does what the user *would want to* do

Instead of writing how-to guides
we write user expectations
into software and protocols,

to automatize all steps a user would need to carry out.

⇒ Taking away "crypto needs" from users view (like *https)*

p≡p
Privacy by Default.

2.1. p≡p Concept: pretty Easy privacy

Makes privacy easy.

Easy to install;
Easy to understand;
Easy to use.

No hassle; No training needed.

Also: Easy for app-devs!

p≡p
Privacy by Default.

2.1. p≡p Concept: Easy: Trustwords

>> Battery Horse Staple <<

instead of

>> EC55 39C8 FECF <<

p≡p

Privacy by Default.

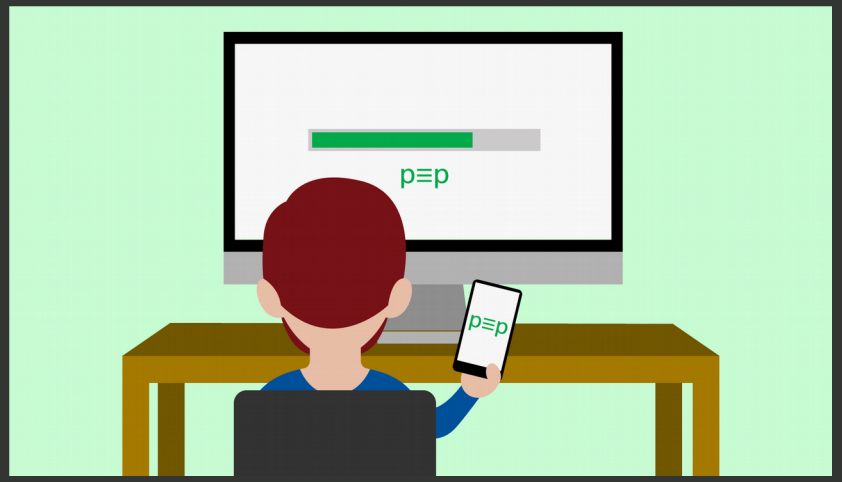
2.1. p≡p Concept: Easy: p≡pSync

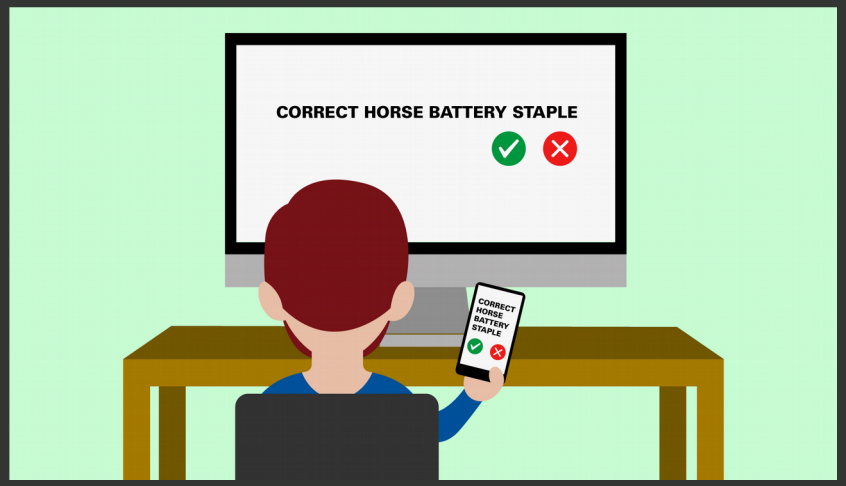
Use same keys on multiple devices:

Realized with the help of Device Groups:

(1) New device generates a device-key,
(2) Pings with this one to the device-group,
(3) Existing devices and user verify the new device,
(4) Devices agree on a secret main group,
(5) All Devices exchange their secret keys.

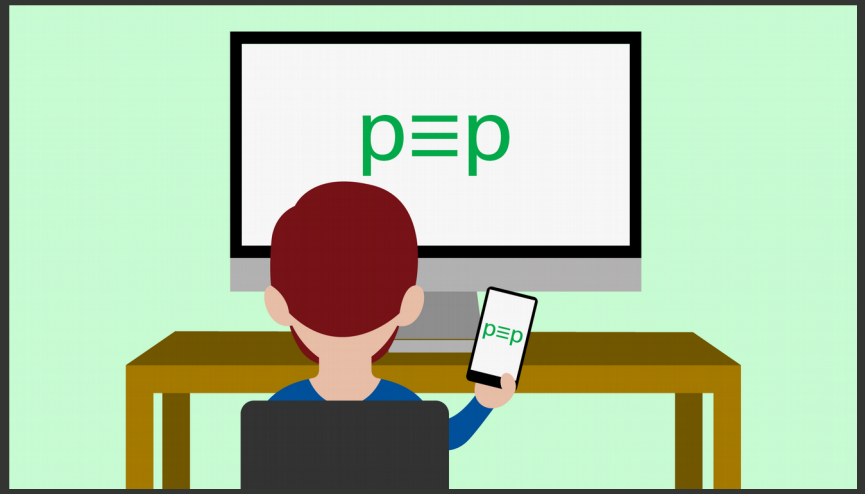
p≡p
Privacy by Default.





Sync keys, contacts and calendar

Realized with device groups – backup-problem solved, too!



p≡p

Privacy by Default.

There is NO CLOUD, just

other people's computers

fsfe.org

2.2. p≡p Concept: End-to-End

End-to-end encryption

Peer-to-peer transport

No centralized infrastructure nor closed services

p≡p

Privacy by Default.

2.3. p≡p Concept: Free Software

p≡p is Free Software

https://pep.foundation/pep-software/

"We publish our code so that our fellow Cypherpunks may practice and play with it. Our code is free for all to use, worldwide."

(Cypherpunk Manifesto, March 1993)

p≡p

Privacy by Default.

2.4. p≡p Concept: Compatibility

Multiple crypto technologies

Multiple message transports

Multiple platforms

Multiple languages

p≡p

Privacy by Default.

2.4. p≡p Concept:Compatib.:Crypto

OpenPGP / GnuPG / netGPG

S/MIME

OTR

OMEMO

Signal Protocol / Axolotl

…

p≡p

Privacy by Default.

2.4. p≡p Concept:Compat:Transports

SMTP / IMAP / POP3 / Exchange

XMPP (jabber)

non-open standards (e.g. Twitter DMs)

GNUnet

SMS

...

p≡p

Privacy by Default.

2.5. p≡p Concept:MetaDataProtection

Content encryption is not everything…

E.g. E-Mail: Metadata stays visible!

(e.g.: from/to, IPs, Subject, size,…)

p≡p encrypts subjects inline (opt-out)
p≡p obfuscates & encrypts the header as much as possible

p≡p
Privacy by Default.

2.5. p≡p Concept:MetaDataProtection

What is the problem with our Internet? e.g..:

Network knows & learns too much

Insecure defaults & high Complexities

Centralized Components
(e.g. IANA, ICANN, DNS, …)

Administrators can be a target!

p≡p
Privacy by Default.

2.5. p≡p Conzept:MetaDataProtection

1970/80: Internet v1.0

Wow, I can access your computer, you can check out mine!
Awesome!

2010/20: Internet v1.1

Sure I can access other computers and use their services.
Wait, What? They can also access mine!?

20XX/25: Internet v2.0

End-to-end encryption and anonymization of the ways data flows.

p≡p

Privacy by Default.

2.6. p≡p Concept: Summary

Users don't have to think about the crypto anymore. They can just use it.

By default.

"It is this 'little hacker inside' that decides on the cryptography chosen to communicate with the message recipient."

p≡p

Privacy by Default.

3. GNUnet

GNUnet.org

Let's make a GNU one!

p≡p

Privacy by Default.

3.0. p≡p Anonymity: GNUnet: Idea

"GNUnet is a **mesh routing layer** for
end-to-end encrypted networking and a framework for
distributed applications **designed to replace the old insecure Internet protocol stack.**"

GNUnet.org

(founded 2002,

followed in academia)

p≡p
Privacy by Default.

3.1. p≡p Anonymity: GNUnet: Layers

(very hard) simplified version of the Internet:

Google, FB & Co

DNS/X.509

TCP/UDP

IP/BGP

Ethernet

Physical Layer

p≡p

Privacy by Default.

3.1. p≡p Anonymity: GNUnet: Layers

Internet:	GNUnet:
Google, FB & Co	...
DNS/X.509	...
TCP/UDP	...
IP/BGP	...
Ethernet	...
Physical Layer	...

p≡p

Privacy by Default.

3.2. p≡p Anonymity: GNUnet: Layers

Internet:	GNUnet:
Google, FB & Co	...
DNS/X.509	...
TCP/UDP	...
IP/BGP	...
Ethernet	...
Physical Layer	HTTPS/TCP/WLAN/...

p≡p

Privacy by Default.

3.2. p≡p Anonymity: GNUnet: Layers

Start with what we have: e.g. TCP, UDP, SMTP, HTTP, HTTPS; WLAN, Bluetooth,…

Unreliable, out-of-order packet delivery semantics.

Automated Transport Selection (ATS) decides.

GNUnet:

…

…

…

…

…

HTTPS/TCP/WLAN/…

p≡p

Privacy by Default.

3.3. p≡p Anonymity: GNUnet: Layers

Internet:	GNUnet:
Google, FB & Co	...
DNS/X.509	...
TCP/UDP	...
IP/BGP	...
Ethernet	CORE (OTR)
Physical Layer	HTTPS/TCP/WLAN/...

p≡p

Privacy by Default.

3.3. p≡p Anonymity: GNUnet: Layers

Off-The-Record encryption between peers.

Multiplexes inbound messages by type to higher-level subsystems.

Hides connections from/to peers that do not speak same higher-level protocol.

GNUnet:

...

...

...

...

CORE (OTR)

HTTPS/TCP/WLAN/...

p≡p

Privacy by Default.

3.4. p≡p Anonymity: GNUnet: Layers

Internet:	GNUnet:
Google, FB & Co	...
DNS/X.509	...
TCP/UDP	...
IP/BGP	R⁵N DHT
Ethernet	CORE (OTR)
Physical Layer	HTTPS/TCP/WLAN/...

p≡p

Privacy by Default.

3.4. p≡p Anonymity: GNUnet: Layers

Decentralized routing algorithm

Using distributed hash tables (randomized version of Kademlia, still effective in small networks)

GNUnet:

...

...

...

R⁵N DHT

CORE (OTR)

HTTPS/TCP/WLAN/...

p≡p

Privacy by Default.

3.5. p≡p Anonymity: GNUnet: Layers

Internet:	GNUnet:
Google, FB & Co	...
DNS/X.509	...
TCP/UDP	CADET
IP/BGP	R⁵N DHT
Ethernet	CORE (OTR)
Physical Layer	HTTPS/TCP/WLAN/...

p≡p

Privacy by Default.

3.5. p≡p Anonymity: GNUnet: Layers

Transport Protocol.
Has features of SCTP and Axolotl;
serves end-to-end-encryption.

Additional services,
eg for pEp:

Xolotl (sphinx+Axolotl)
protecting meta data,

Lake (like pond) providing
mailboxes / asynchronous
delivery.

GNUnet:

...

...

CADET

R⁵N DHT

CORE (OTR)

HTTPS/TCP/WLAN/...

p≡p
Privacy by Default.

3.5. p≡p Anonymity: GNUnet: Layers

by all science & mathematics we know today,

all the meta data will be gone that way!

<u>GNUnet:</u>

...

...

CADET

R⁵N DHT

CORE (OTR)

HTTPS/TCP/WLAN/...

p≡p

Privacy by Default.

3.6. p≡p Anonymity: GNUnet: Layers

Internet:	GNUnet:
Google, FB & Co	...
DNS/X.509	GNS
TCP/UDP	CADET
IP/BGP	R⁵N DHT
Ethernet	CORE (OTR)
Physical Layer	HTTPS/TCP/WLAN/...

p≡p

Privacy by Default.

3.6. p≡p Anonymity: GNUnet: Layers

Secure and decentralized name system, no central root zones or auth.

Provides alternative public key infrastructure.

Inter-operable with DNS.

Query and response privacy.

GNUnet:

...

GNS

CADET

R⁵N DHT

CORE (OTR)

HTTPS/TCP/WLAN/...

p≡p

Privacy by Default.

3.7. p≡p Anonymity: GNUnet: Layers

Internet:	GNUnet:
Google, FB & Co	Applications
DNS/X.509	GNS
TCP/UDP	CADET
IP/BGP	R⁵N DHT
Ethernet	CORE (OTR)
Physical Layer	HTTPS/TCP/WLAN/...

p≡p

Privacy by Default.

3.7. p≡p Anonymity: GNUnet: Layers

File sharing

SecuShare (social networking)

Conversation (VoIP)

pEp (messaging)

GNU Taler (payments)

MUDs (game)

Your app?

GNUnet:

Applications

GNS

CADET

R⁵N DHT

CORE (OTR)

HTTPS/TCP/WLAN/...

p≡p

Privacy by Default.

3.8. p≡p Anonymity: GNUnet: Goals

"GNUnet wants to...

...protect the privacy of its users and to guard itself against attacks or abuse.

…become a widely used, reliable, open, non-discriminating, egalitarian, unfettered and censorship-resistant system of free information exchange.

…serve as a development platform for the next generation of decentralized Internet protocols."

p≡p
Privacy by Default.

3.9. p≡p Try GNUnet!

Check it out!

Clone gnunet.org/git
Follow instructions on the website

Get support via #gnunet on freenode
and/or via ML e.g. help-gnunet@gnu.org

! Report bugs on gnunet.org/bugs !

Written in C, but a GNUnet-Java exist, too:
Start for an API for extensions in Java :)

p≡p
Privacy by Default.

Summary: Problem & Solution

Problem:

Online communication is visible like a postcard &
this world has mass surveillance

Solution:

Right now: Mass encryption
Then: Mass anonymization

p≡p
Privacy by Default.

Summary: Human Rights

Article 12

"No one shall be subjected to arbitrary interference with his privacy, family, home or correspondence, nor to attacks upon his honour and reputation.

Everyone has the right to the protection of the law against such interference or attacks."

http://www.un.org/en/universal-declaration-human-rights/

Summary: Law of Mathematics

The laws of Australia will trump the laws of mathematics: Turnbull

Despite calling the laws of mathematics 'commendable', the prime minister of Australia told ZDNet the only law that applies in Australia is the law of Australia when it comes to legislating decryption.

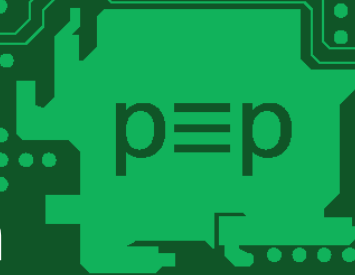
By Chris Duckett and Asha McLean | July 14, 2017 -- 01:27 GMT (02:27 BST) | Topic: Security

"The laws of Australia prevail in Australia, I can assure you of that," he said on Friday. "The laws of mathematics are very commendable, but the only law that applies in Australia is the law of Australia."

Questions?

pretty Easy privacy:

#prettyeasyprivacy on Freenode
@pEpfoundation on twitter
#prettyeasyprivacy
https://pEp.foundation/
https://pEp.software/
https://pEp.community/



Speaker:

sva@pEp.foundation
sva@IRC (various networks)
@sva on twitter