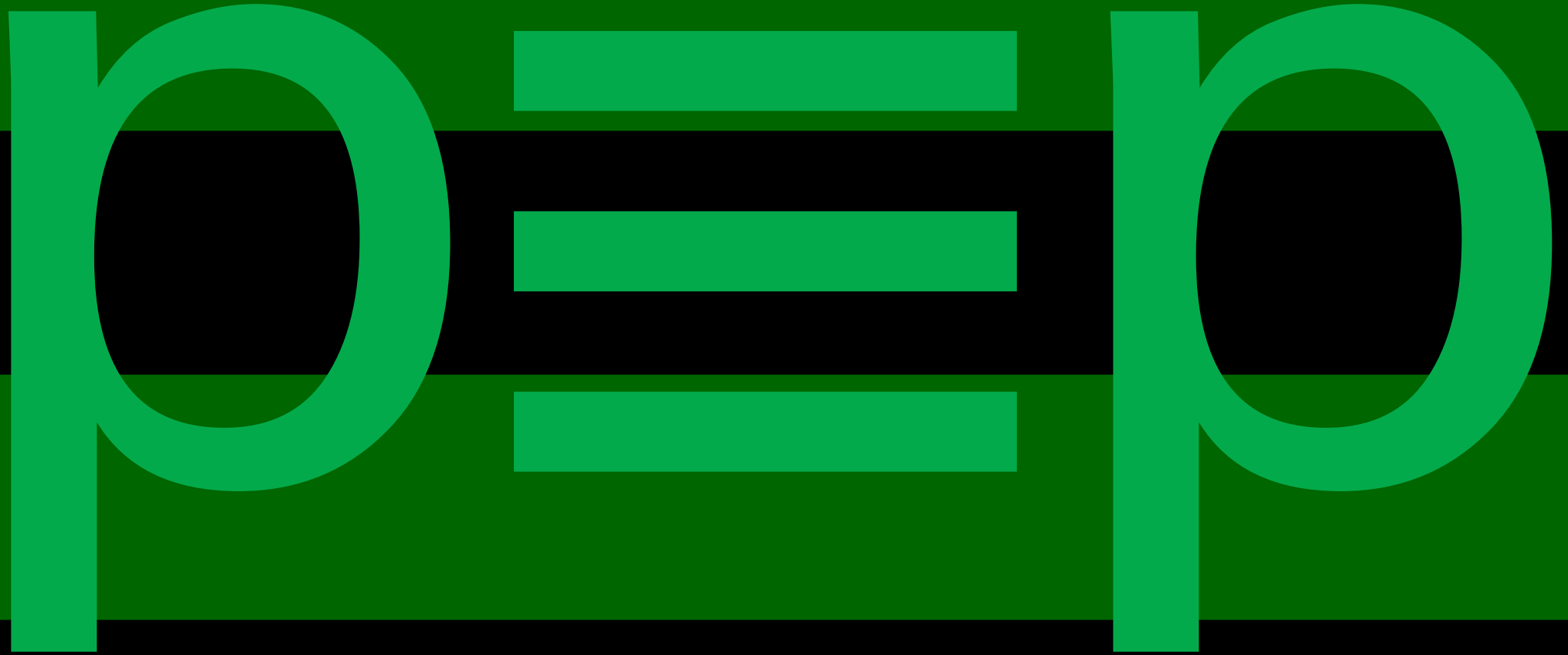


pretty Easy privacy



sva@pEp.foundation <https://pEp.foundation>

twitter@sva twitter@pEpfoundation

Privacy by Default.

Überblick

- 0 – Intro
- 1 – Konzept
- 2 – Organisation
- 3 – Technologie
- 4 – Derzeitige Implementierung

0 – Intro

Problem:

Online Kommunikation ist wie eine Postkarte
& diese Welt hat Massenüberwachung

Lösung:

Zuerst: Massenverschlüsselung

Später: Massenanonymisierung



1 – Konzept: Überblick

1.0. Privacy by Default

1.1. pretty Easy privacy

1.2. Peer-to-Peer und Ende-zu-Ende

1.3. Freie Software

1.4. Kompatibilität (Crypto & Transports)

1.5. Anonymität (GNet)

pep

1.0. p≡p Konzept: Privacy by Default

Privacy by Default.

p≡p macht das, was der Nutzer eigentlich tun sollte... (oder hätte tun sollen ;))

Anstatt immer mehr Anleitungen zu schreiben, schreiben wir jetzt eben Protokolle & Software



1.1. p≡p Konzept:: pretty Easy privacy

pretty Easy privacy by Default.

Einfach zu installieren,
einfach zu verstehen,
einfach zu benutzen.

Verschlüsselung als Standard

Also: Easy für App-Devs



1.1. Easy: Trustwords

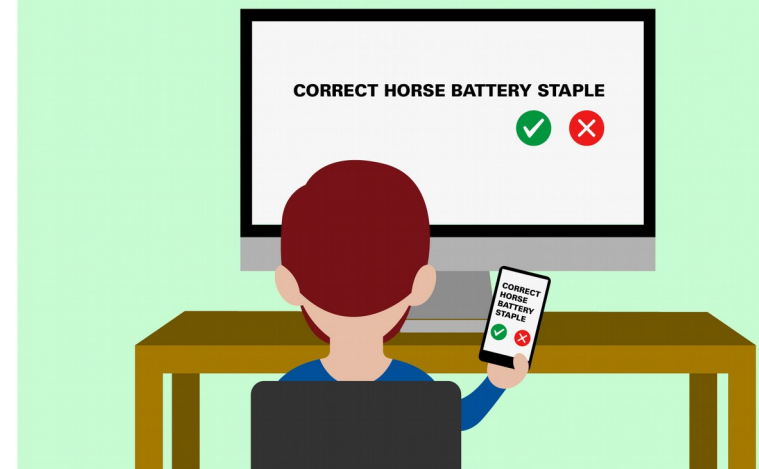
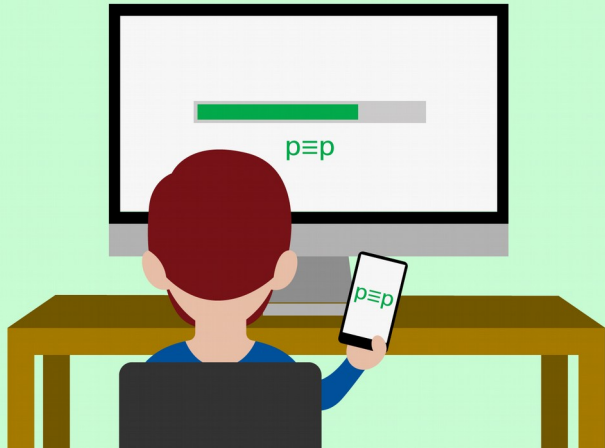
>> **Battery Horse Staple** <<

anstatt

>> **EC55 39C8 FECE** <<

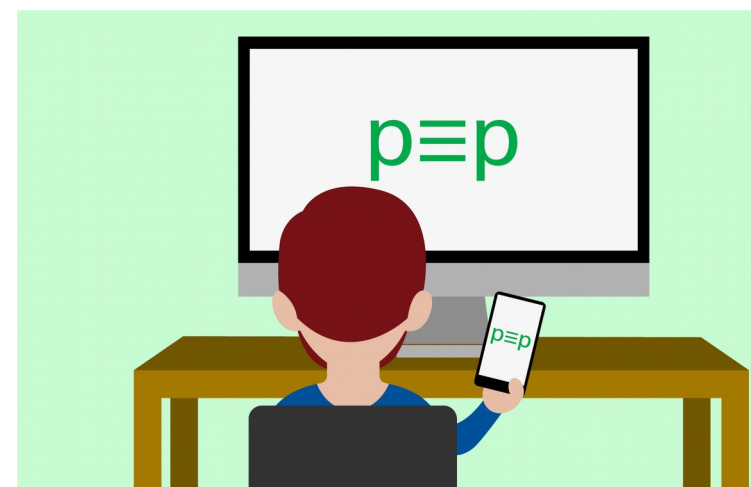


1.1. Easy: $p \equiv p$ Sync

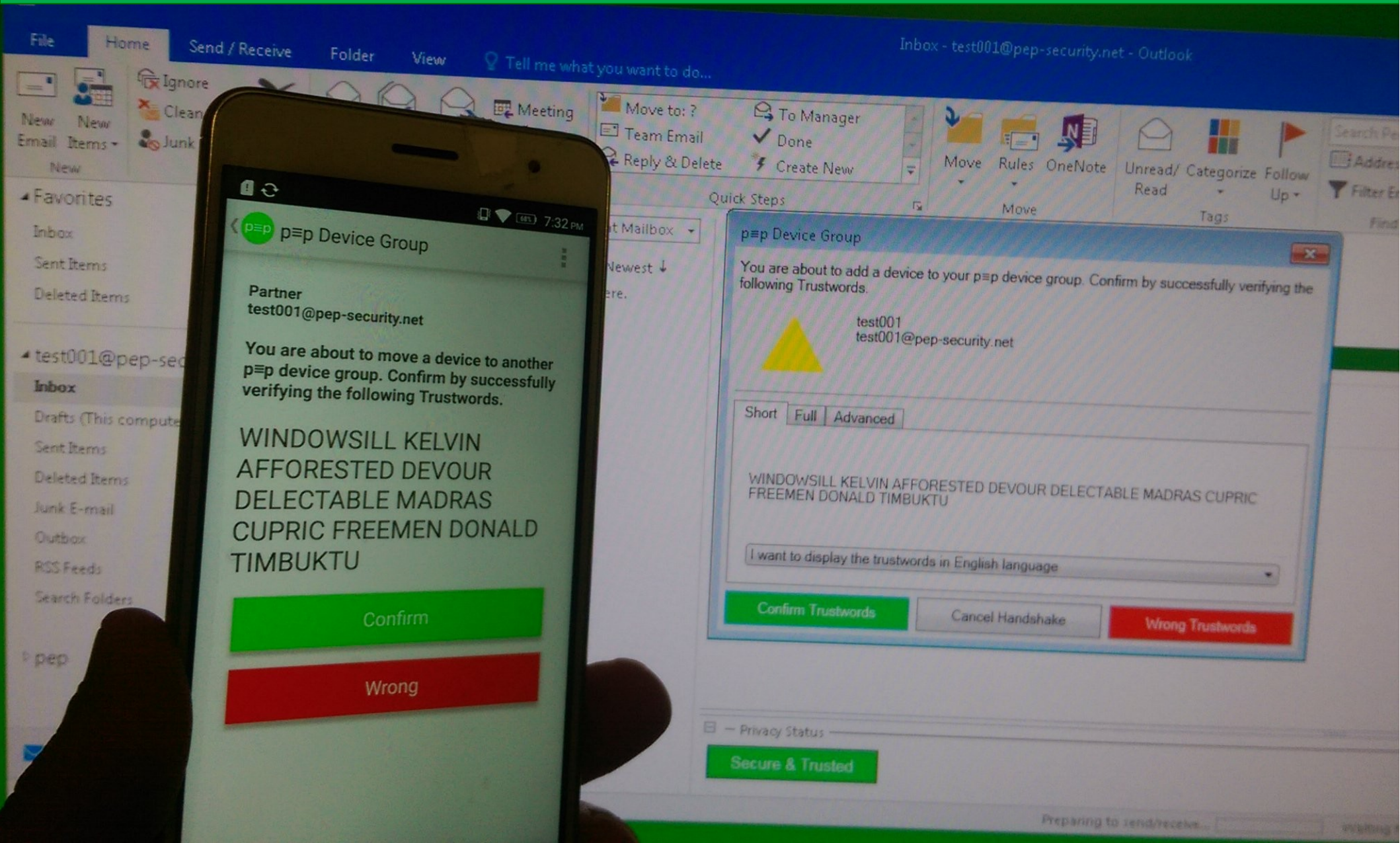


Sync keys, contacts and calendar

Realisiert mit Gerätegruppen – backup-problem solved, too!



1.1. Summary: Trustwords & p≡pSync



1.2. p≡p Konzept: Peer-to-Peer

Peer-to-peer Transport

Ende-zu-Ende Verschlüsselung

Keine zentrale Infrastruktur oder
geschlossene Services

p≡p

1.3. p≡p Konzept: Free Software

p≡p ist Freie Software,

<https://cacert.pep.foundation/trac/>

<https://letsencrypt.pep.foundation/trac>

(GPLv3)

Es werden laufend externe Code Reviews durchgeführt und veröffentlicht



1.4. $p \equiv p$ Konzept: Kompatibilität

Möglichst viele Plattformen

Möglichst viele Sprachen

Möglichst viele Verschlüsselungstechnologien

Möglichst viele Nachrichtenübermittlungsprotokolle
(message transport protocols)



1.4. $p \equiv p$ Konzept: Komp.: Crypto

OpenPGP / GnuPG

S/MIME

OTR

OMEMO

Signal Protocol / Axolotl

...

$p \equiv p$

1.4. $p \equiv p$ Konzept: Komp.:Transports

SMTP / IMAP / POP3 / Exchange

XMPP (jabber)

Nicht-offene Transporte
(e.g. Twitter DMs)

SMS

...

$p \equiv p$

1.5. $p \equiv p$ Konzept: Anonymität

Verschlüsselung ist nicht alles...

Bsp. E-Mail:

Metadaten bleiben trotzdem sichtbar!

(z.B: von/an, IPs, Umfang, Betreff,...)

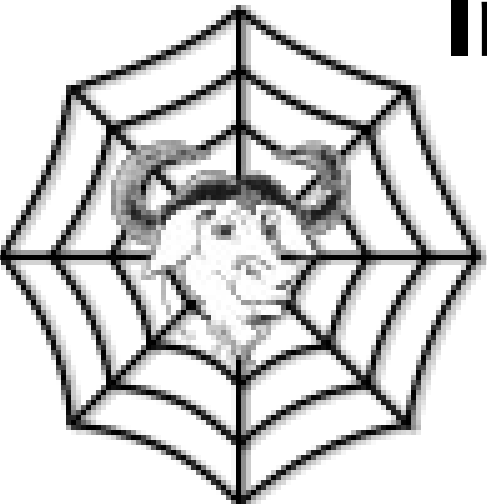
Betreff wird bei $p \equiv p$ inline verschlüsselt (opt-out)



1.5. p≡p Konzept: Anonymity/GNUnet

“You broke the Internet, lets make a GNU one!”

“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)

p≡p

1.6. $p \equiv p$ Konzept: Zusammenfassung

Nutzer müssen nicht über die Verschlüsselung nachdenken, sondern können es einfach nutzen.

By default.

“Es ist also ein kleiner Hacker da drinnen, der entscheidet, welche Verschlüsselung dem Empfänger der Nachricht am besten passt.”

$p \equiv p$

2 – Organisation

Firma:

<https://prettyeasyprivacy.com/>
Verkauft Anwendungen und Services

Stiftung:

<https://pep.foundation/>
Unterstützt Freie Software
Code gehört der Stiftung!



3 – Technologie: Übersicht

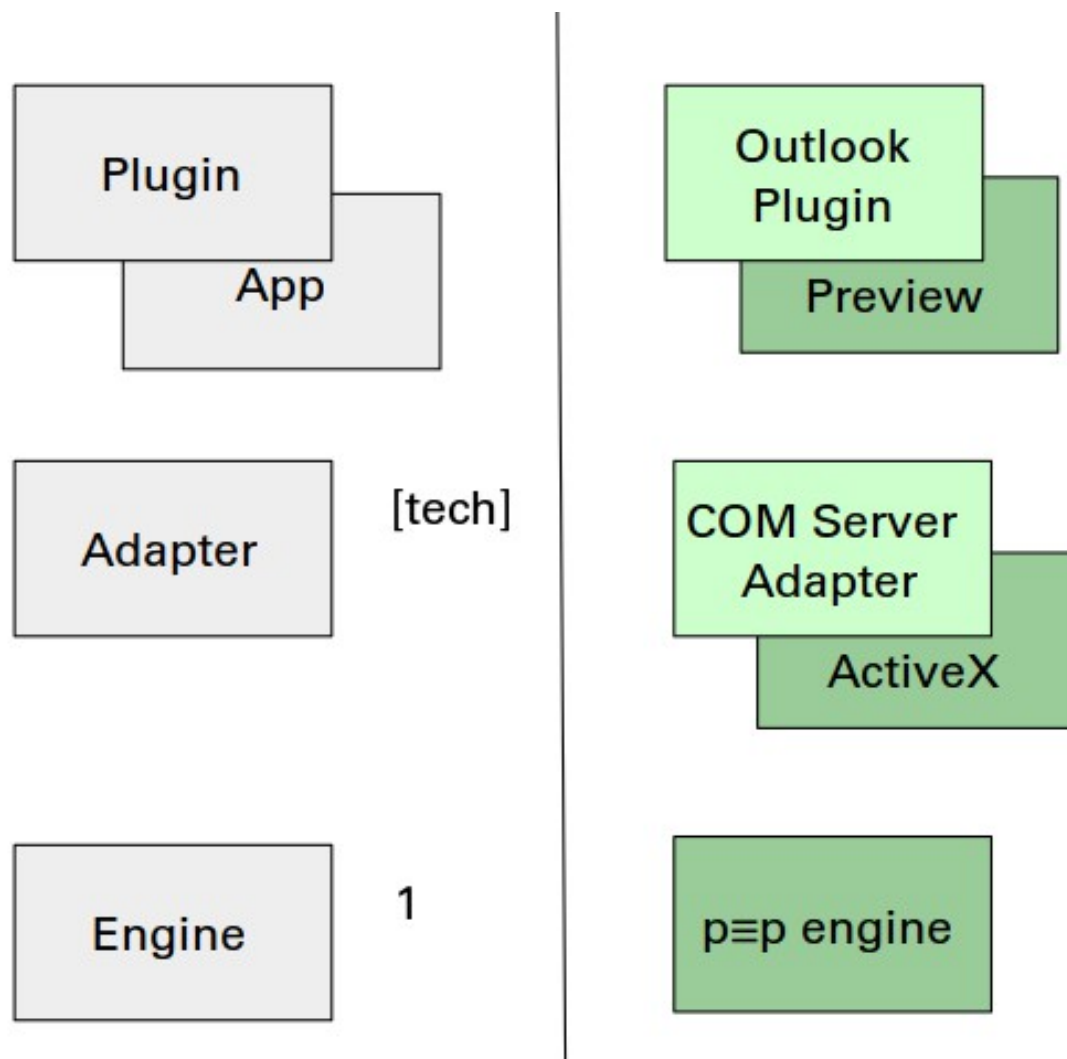
3.0. Architektur

3.1. Engine und Adapter

3.2. Liste der Adapter und Repos

3.3. Liste der Developing Platforms

3.0. p≡p Tech: Architektur



Anwendungen

Adapter

und Engine



3.2. p≡p Tech: Adapters & Repos

MailModel	modelling Message and Folder
netpgp-et	fork of netpgp (iOS adaptations 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
pEpPythonAdapter	p≡p Python adapter
pEpQtAdapter	p≡p Qt adapter
pEpiOSAdapter	p≡p iOS adapter
pantomime-iOS	fork of pantomime (iOS adaptations)
yml2	>b's YML 2

<https://cacert.pep.foundation/dev>

<https://letsencrypt.pep.foundation/dev>



3.3. p≡p Tech: Entwicklungsplattform

iOS

Android

Linux

BSD

MacOS

Windows



4 – Applications: Übersicht

4.0. Derzeitige Implementierung

4.1. Android via K-9-Fork

4.2. MS Outlook via Add-in

4.3. Thunderbird via Enigmail/p≡p



4.0. p≡p Apps: Implementatierung

Macht PGP/GPG und S/MIME ohne den Nutzer zu stören.

Verschlüsselt automatisch

Kein Key Management nötig

Kein Keyserver oder andere zentrale Infrastruktur

Fingerprints ≡ Trustwords

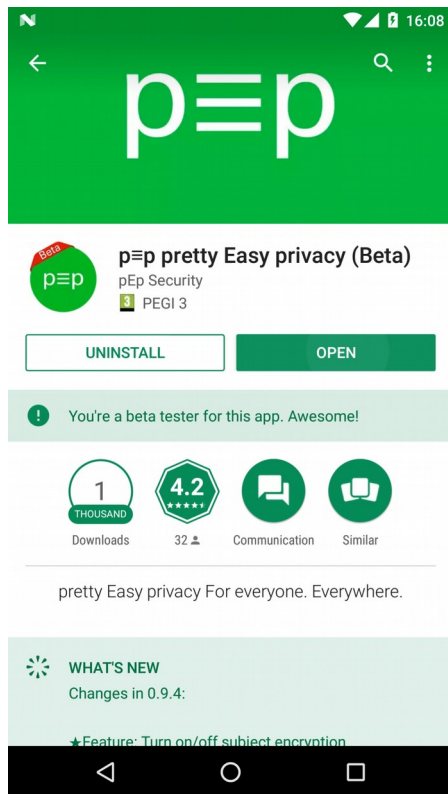
Keine Passphrases

Header verschlüsselt und verschleiern

p≡pSync



4.1. p≡p Apps: Android/K-9-Fork



Ready to use

Verfügbar auf
Play Store and F-Droid



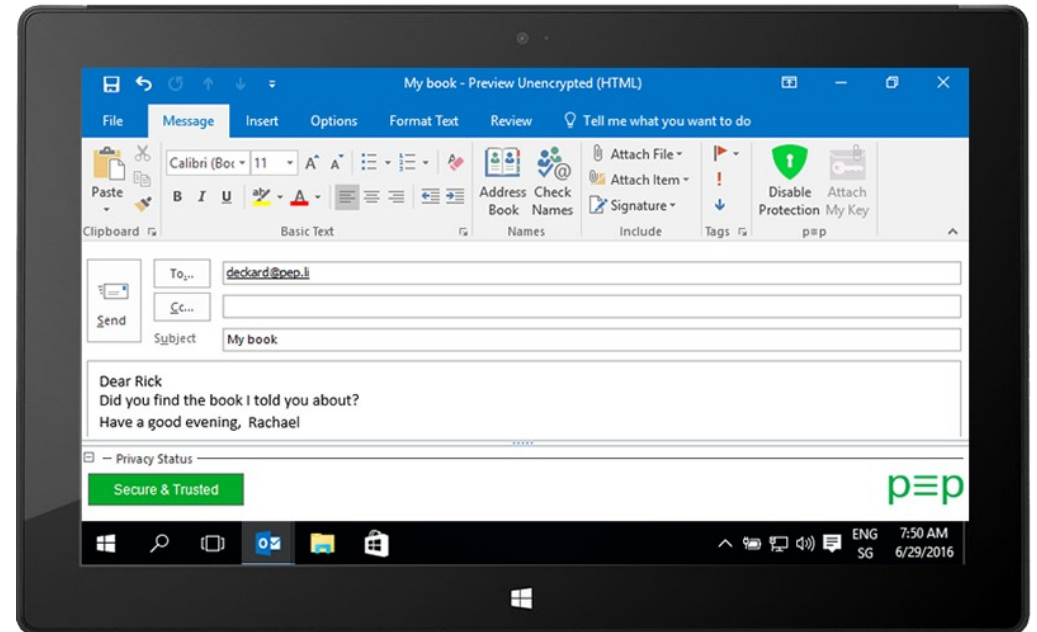
4.2. p≡p Apps: Windows/Outlook

Ready to use

Verfügbar auf
prettyeasyprivacy.com

Free as in freedom
not as in free beer

(Contact us for testing license)



4.3. p≡p Apps: Thunderbird/Enigmail

Kommt mit dem
nächsten Release of Enigmail

Kein Key Management mehr
(wenn du das möchtest)

(alter Modus bleibt als *expert mode* erhalten)



Fragen?

pretty Easy privacy:

#prettyeasyprivacy on Freenode

twitter@pEpfoundation

<https://pEp.foundation/>

<https://pEp-project.org/>

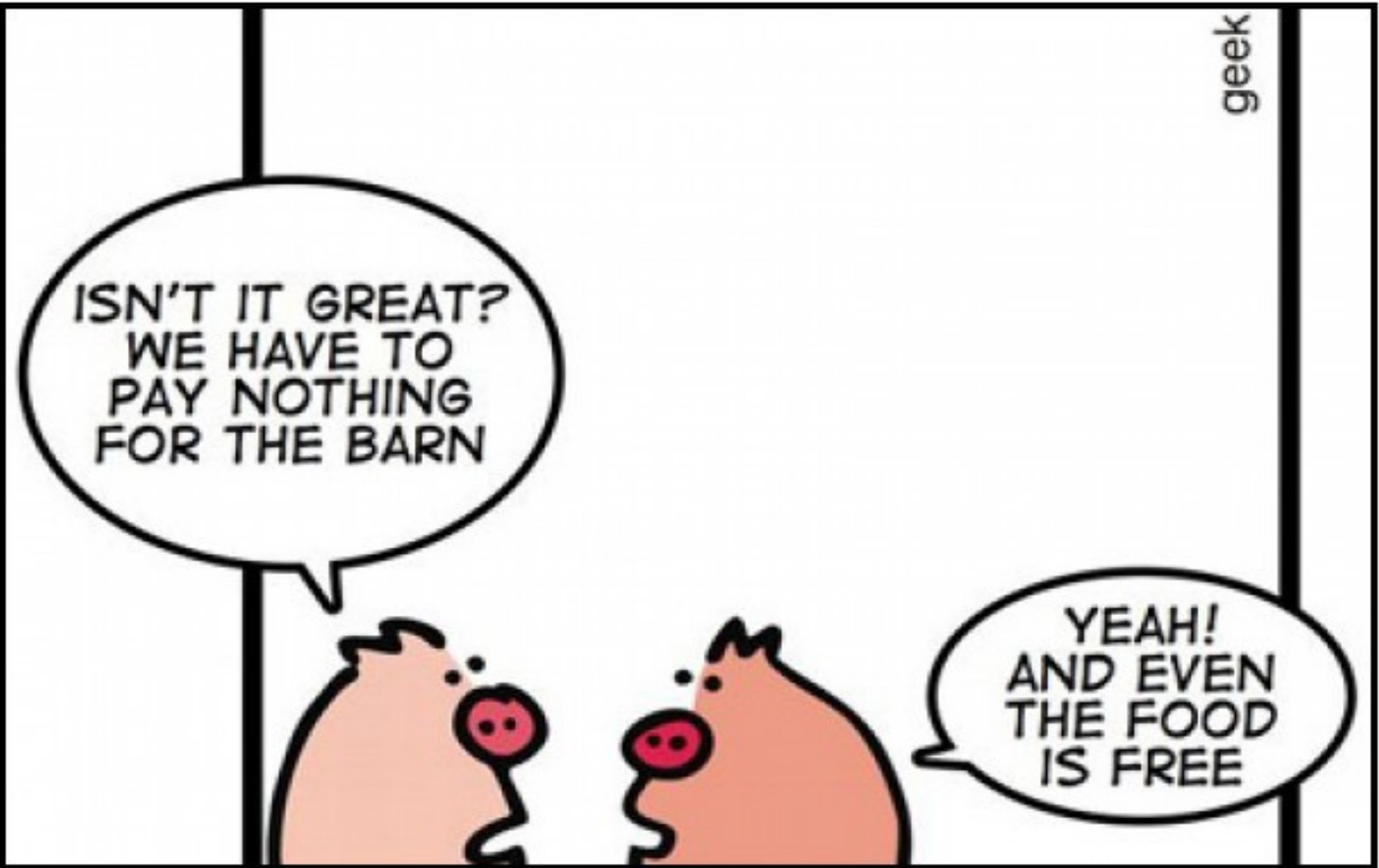
Speaker:

sva@pEp.foundation

sva@IRC (various networks)

twitter@sva

p≡p



ISN'T IT GREAT?
WE HAVE TO
PAY NOTHING
FOR THE BARN

YEAH!
AND EVEN
THE FOOD
IS FREE

FACEBOOK AND YOU

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