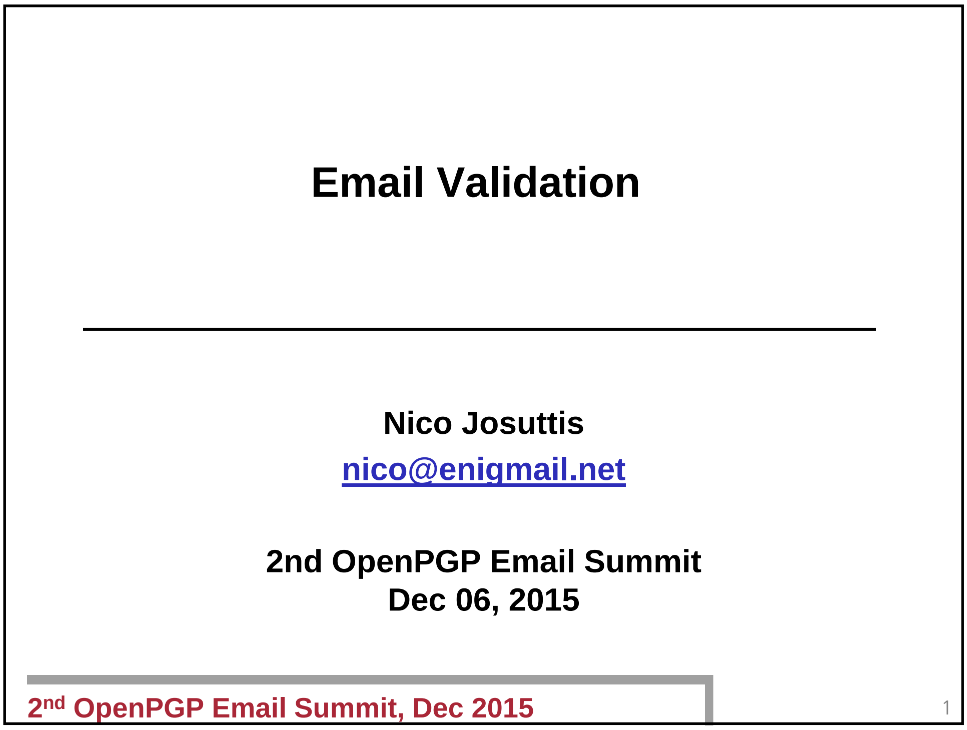
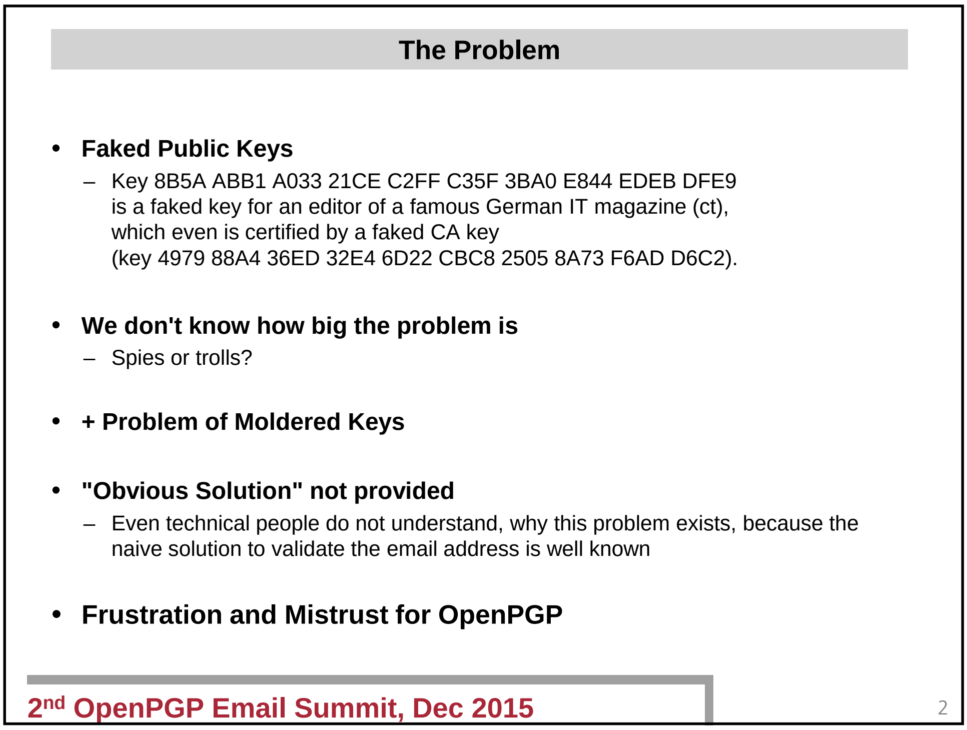


Email Validation

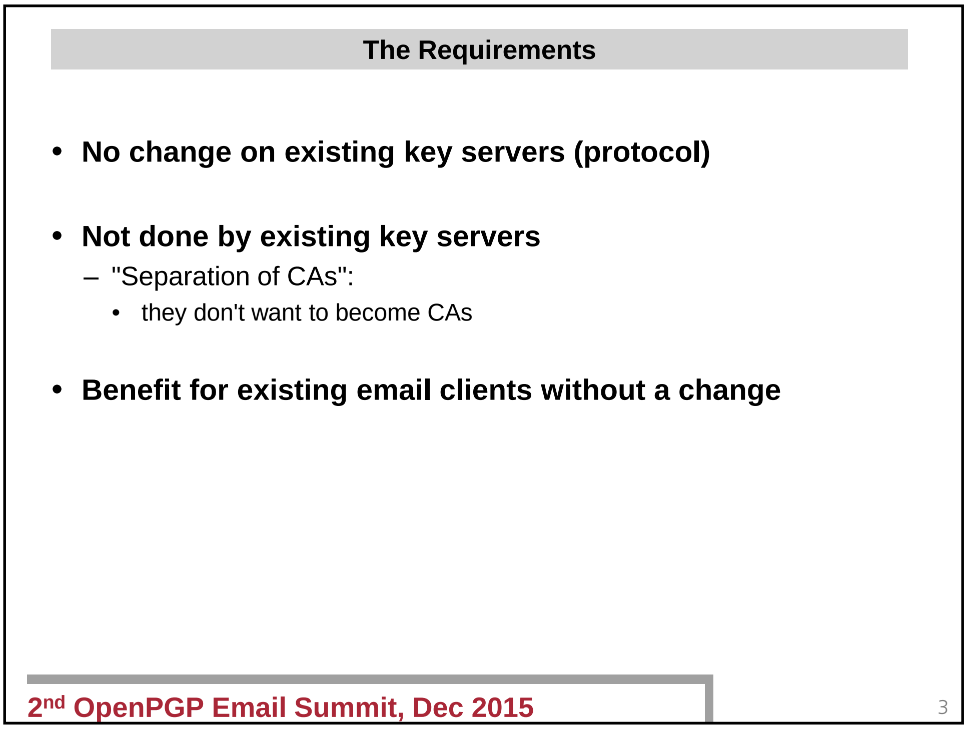
Nico Josuttis
nico@enigmail.net

2nd OpenPGP Email Summit
Dec 06, 2015



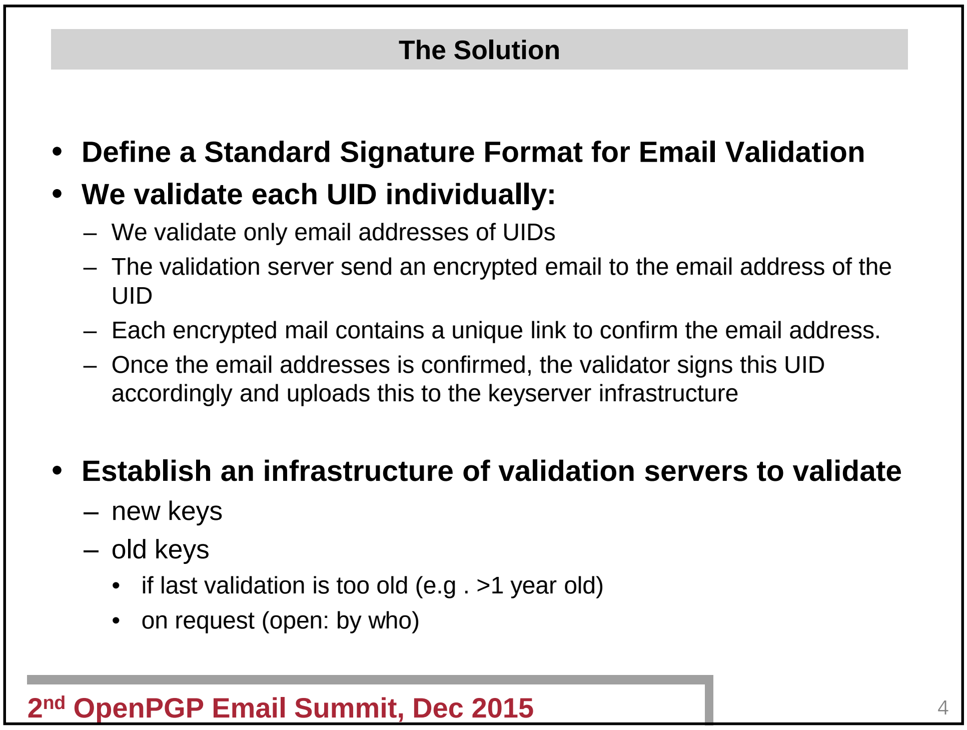
The Problem

- **Faked Public Keys**
  - Key 8B5A ABB1 A033 21CE C2FF C35F 3BA0 E844 EDEB DFE9 is a faked key for an editor of a famous German IT magazine (ct), which even is certified by a faked CA key (key 4979 88A4 36ED 32E4 6D22 CBC8 2505 8A73 F6AD D6C2).
- **We don't know how big the problem is**
  - Spies or trolls?
- **+ Problem of Moldered Keys**
- **"Obvious Solution" not provided**
  - Even technical people do not understand, why this problem exists, because the naive solution to validate the email address is well known
- **Frustration and Mistrust for OpenPGP**



The Requirements

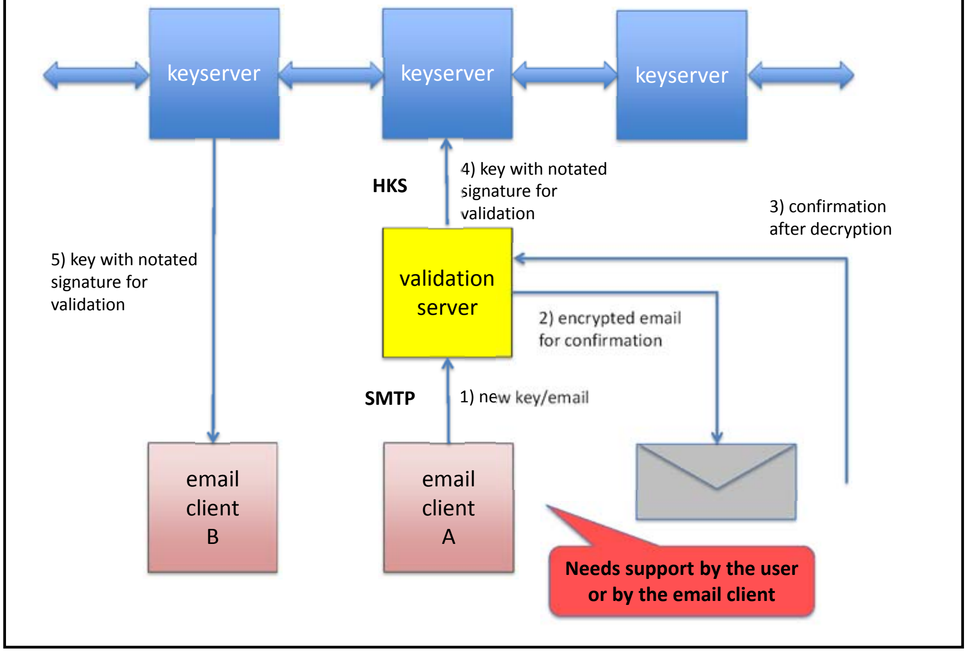
- **No change on existing key servers (protocol)**
- **Not done by existing key servers**
  - "Separation of CAs":
    - they don't want to become CAs
- **Benefit for existing email clients without a change**



The Solution

- **Define a Standard Signature Format for Email Validation**
- **We validate each UID individually:**
  - We validate only email addresses of UIDs
  - The validation server send an encrypted email to the email address of the UID
  - Each encrypted mail contains a unique link to confirm the email address.
  - Once the email addresses is confirmed, the validator signs this UID accordingly and uploads this to the keyserver infrastructure
- **Establish an infrastructure of validation servers to validate**
  - new keys
  - old keys
    - if last validation is too old (e.g . >1 year old)
    - on request (open: by who)

Option 1: Email client has to request for validation



keyserver

keyserver

keyserver

HKS

4) key with notated signature for validation

3) confirmation after decryption

5) key with notated signature for validation

validation server

2) encrypted email for confirmation

SMTP

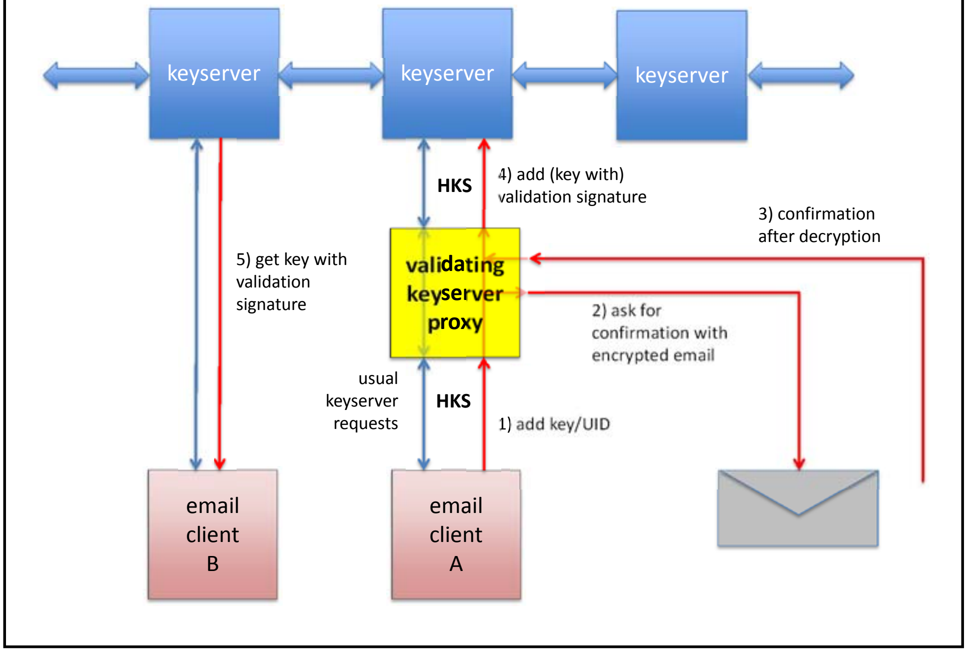
1) new key/email

email client B

email client A

Needs support by the user or by the email client

Option 2: Proxy Server as Key Server Proxy



keyserver

keyserver

keyserver

HKS

4) add (key with) validation signature

3) confirmation after decryption

5) get key with validation signature

validating keyserver proxy

2) ask for confirmation with encrypted email

usual keyserver requests

HKS

1) add key/UID

email client B

email client A

Signature Notations

```
$ gpg2 --charset utf-8 --display-charset utf-8 --check-sigs --
   list-options show-notations,show-policy-urls,show-sig-expire
   0x0B7F8B60E3EDFAE3
```

```
pub   4096R/E3EDFAE3 2007-12-15 [expires: 2016-12-31]
...
uid              Kristian Fiskerstrand <kristian.fiskerstrand@sumptuouscapital.com>
...
sig!3        E3EDFAE3 2013-11-03 never       Kristian Fiskerstrand
   <kristian.fiskerstrand@sumptuouscapital.com>
sig!2   PNX  08AB4849 2014-02-08 2015-02-08  Niels Laukens
   Signature policy: http://niels.dest-unreach.be/pgp-key-signing-policy.txt
   Signature notation: occasion@niels.dest-unreach.be="Zimmermannâ€“Sassaman based key
   signing party at FOSDEM2014 on 2014-02-02"
```

Signature Notation Key

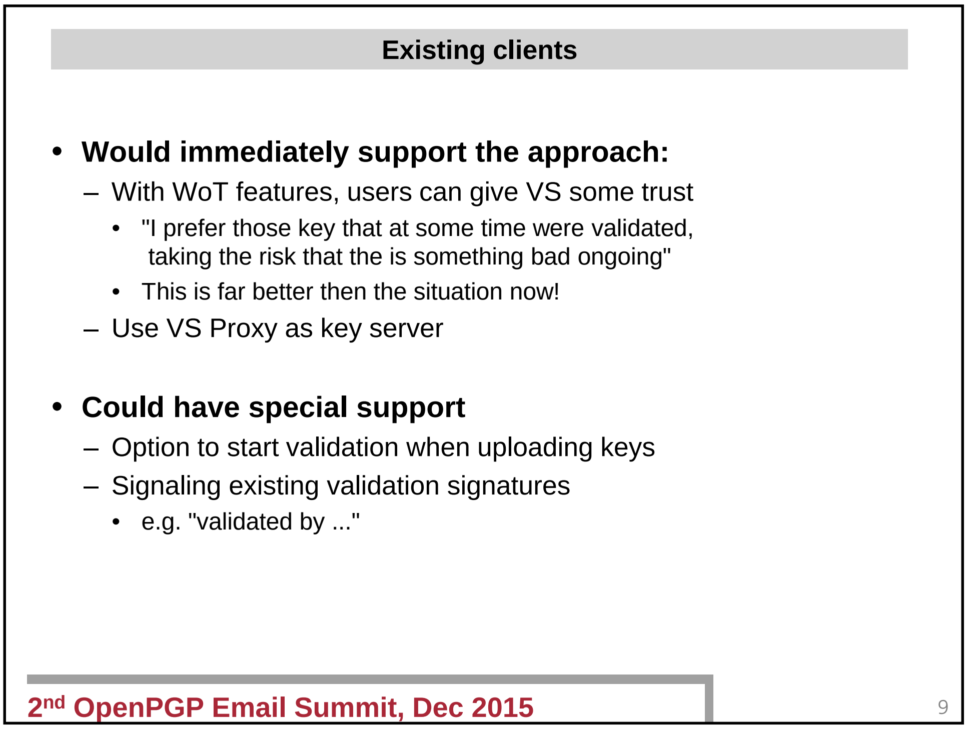
Signature Notation Value

Proposed Signature Format

- **Standardized Signature Notation Key:**
  - e.g. "validation@enigmail.net"
- **Standardized Signature Notation Value Format (open for extension):**
  - Base64 encoded JSON with e.g. the f:

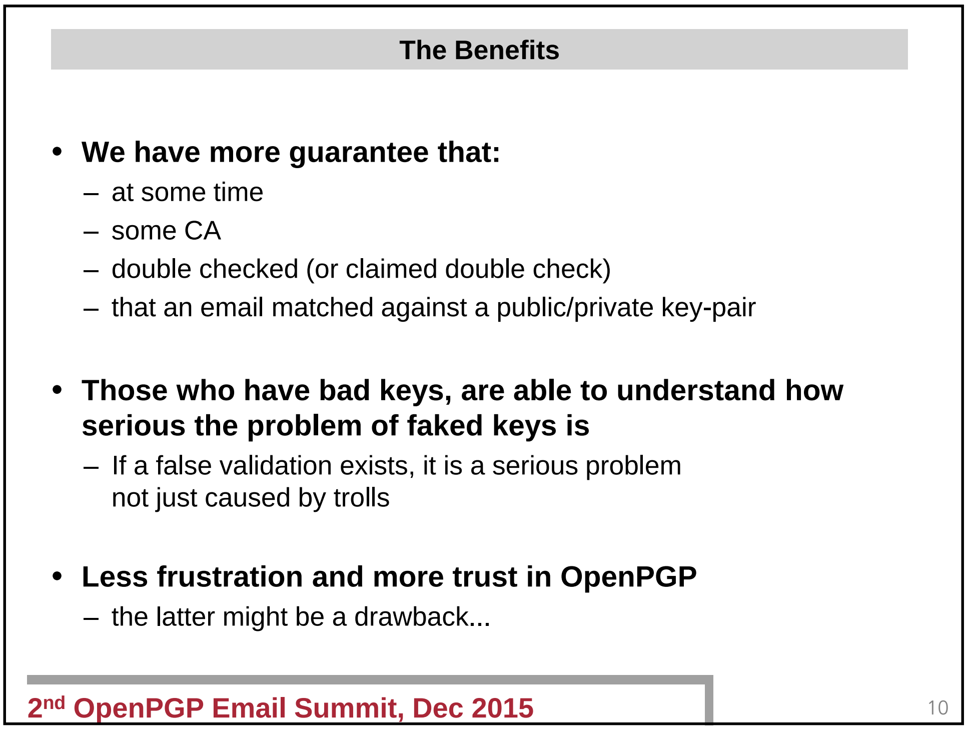
```
{"validation":
 {"validations": [
   { "date": "2014-12-31",
     "type": "enc-email",
     "email": "nico@josuttis.de" },
  ]
 }
}
```

- **Certification check level, “cert-level”:**
  - casual checking (sig2)
- **Expires after 1 year**



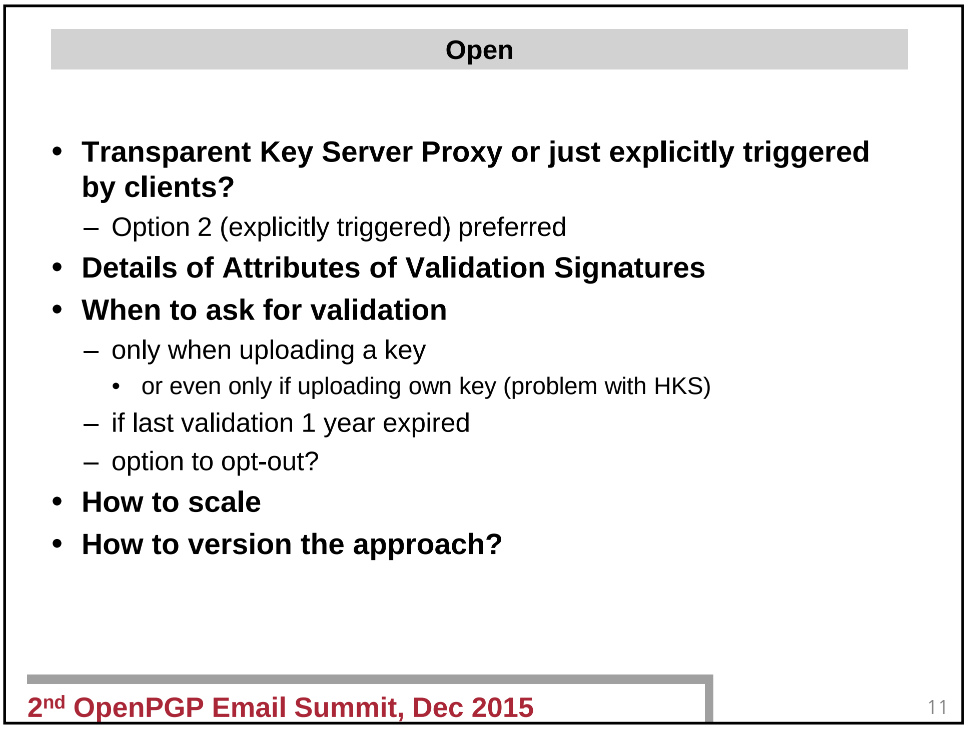
Existing clients

- **Would immediately support the approach:**
  - With WoT features, users can give VS some trust
    - "I prefer those key that at some time were validated, taking the risk that the is something bad ongoing"
    - This is far better then the situation now!
  - Use VS Proxy as key server
- **Could have special support**
  - Option to start validation when uploading keys
  - Signaling existing validation signatures
    - e.g. "validated by ..."



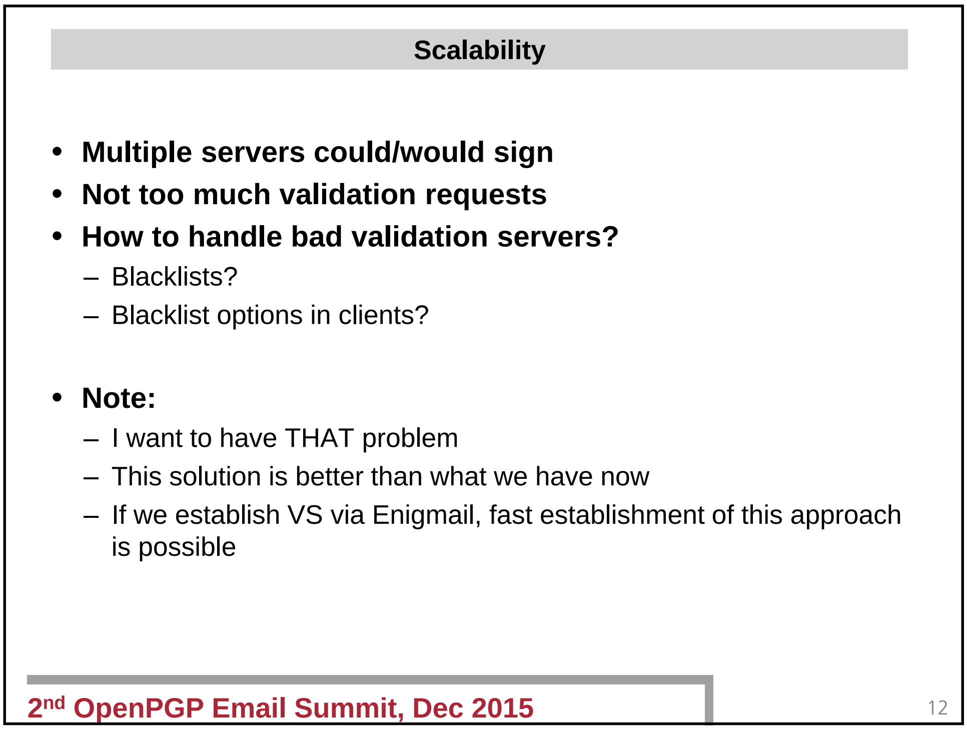
The Benefits

- **We have more guarantee that:**
  - at some time
  - some CA
  - double checked (or claimed double check)
  - that an email matched against a public/private key-pair
- **Those who have bad keys, are able to understand how serious the problem of faked keys is**
  - If a false validation exists, it is a serious problem not just caused by trolls
- **Less frustration and more trust in OpenPGP**
  - the latter might be a drawback...



Open

- **Transparent Key Server Proxy or just explicitly triggered by clients?**
  - Option 2 (explicitly triggered) preferred
- **Details of Attributes of Validation Signatures**
- **When to ask for validation**
  - only when uploading a key
    - or even only if uploading own key (problem with HKS)
  - if last validation 1 year expired
  - option to opt-out?
- **How to scale**
- **How to version the approach?**



Scalability

- **Multiple servers could/would sign**
- **Not too much validation requests**
- **How to handle bad validation servers?**
  - Blacklists?
  - Blacklist options in clients?

- **Note:**
  - I want to have THAT problem
  - This solution is better than what we have now
  - If we establish VS via Enigmail, fast establishment of this approach is possible