GPG & E2EE

—— 论在数字时代，什么才是可以信任的

RSA

- 私玥可以导出公钥；公钥无法反推私钥
- 公钥加密数据 => 仅有私钥可以解密
- 私钥签名数据 => 可以使用公钥验签

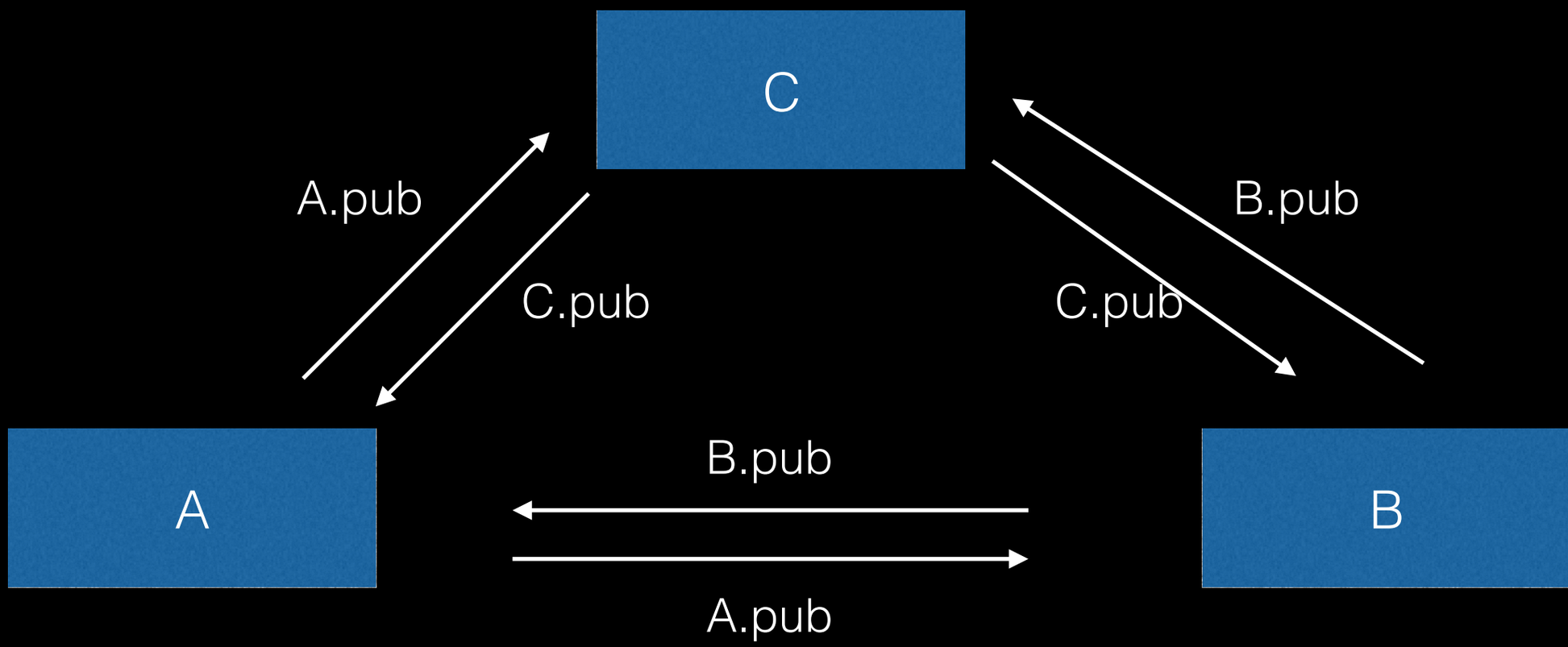
密码学账户

- SSL Certificate
- S/MIME
- SSH Key
- Bitcoin Wallet
- GPG Identity

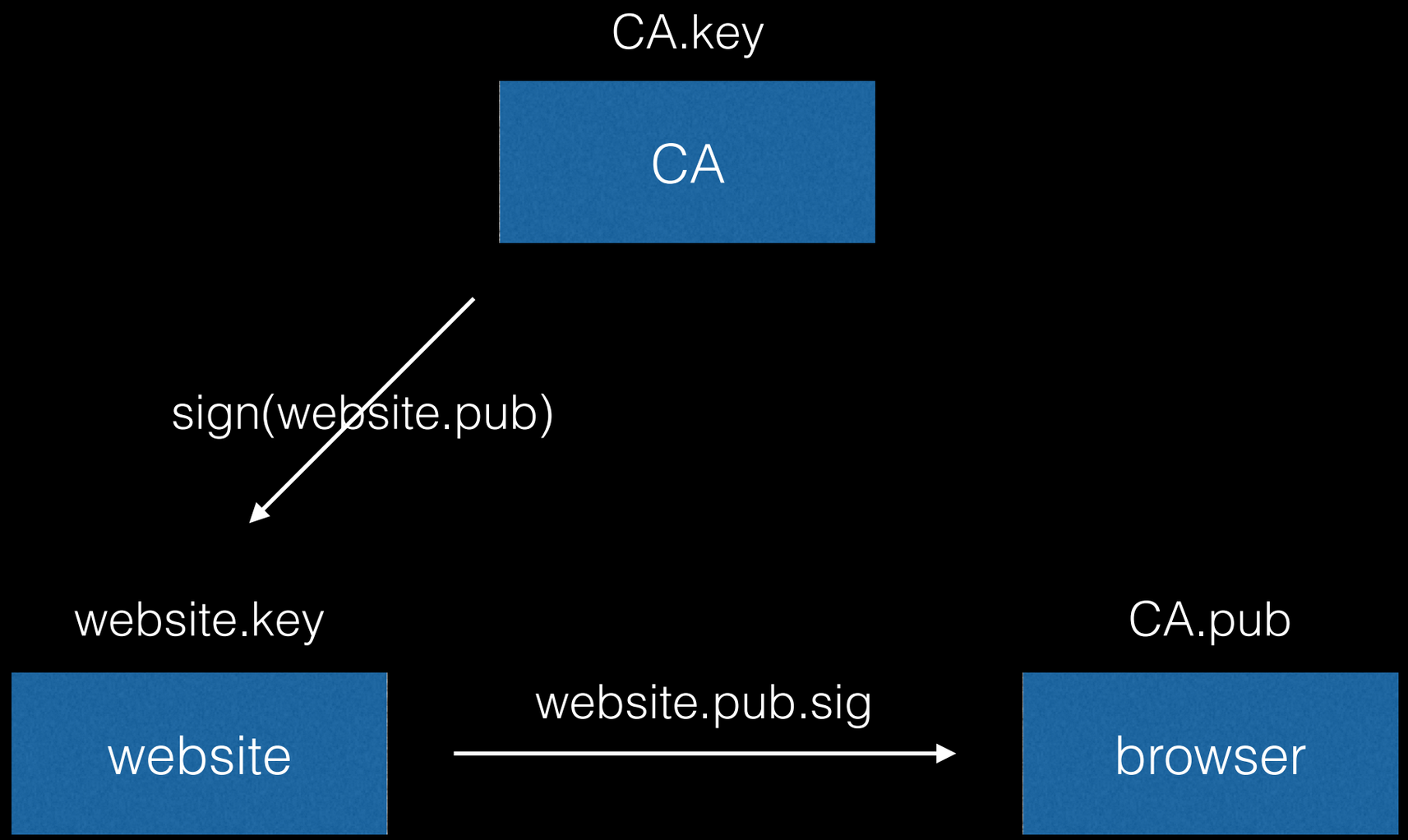
「公钥交换是一个永远绕不开的问题」

– Wang Ziting

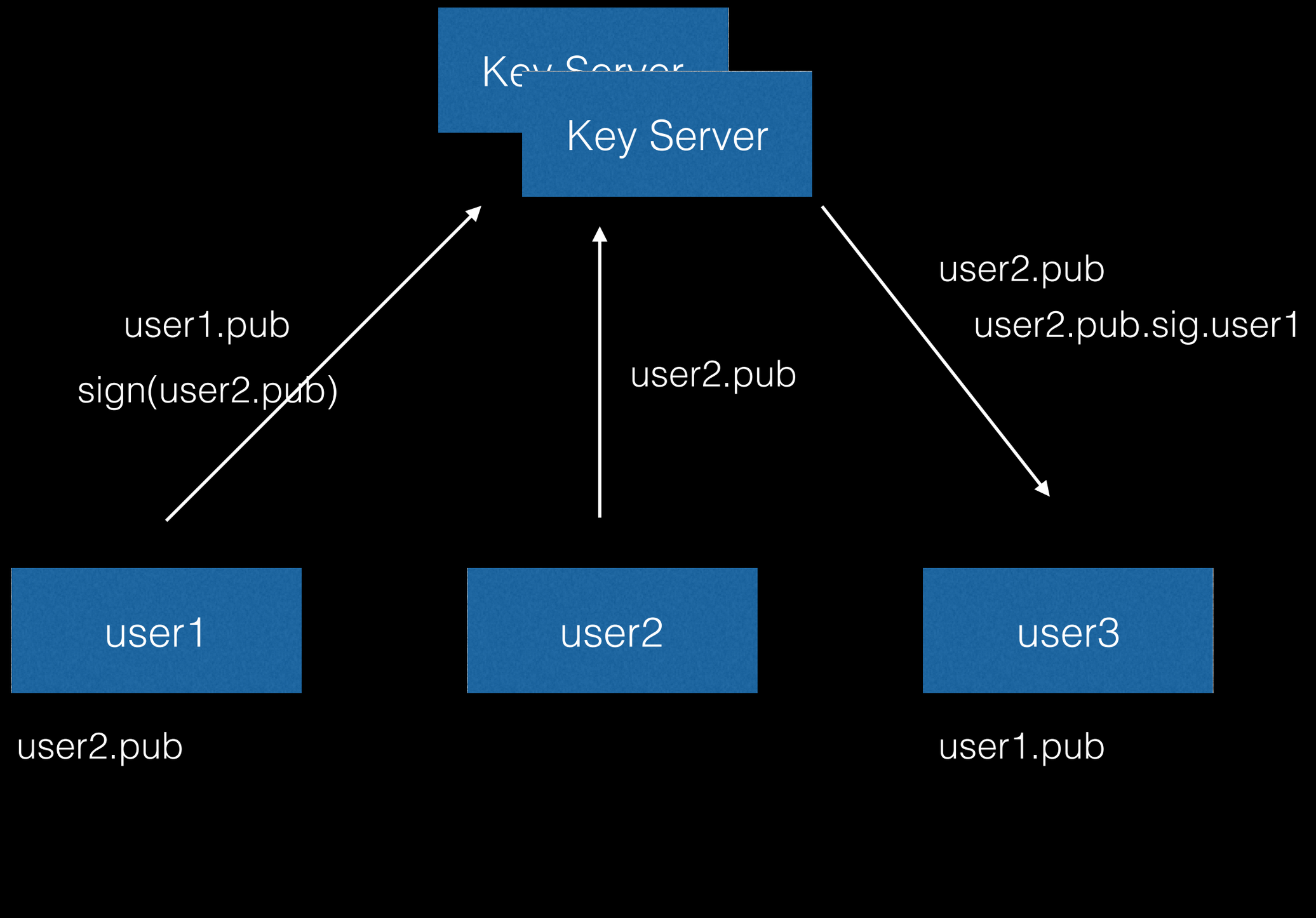
中间人攻击



X.509 Certificate chains



GPG Trust Model



Key Signing Party

```
jysperm@jysperm-MacBook-Pro ~> gpg --list-sigs jysperm
pub   4096R/E466CF1E 2014-11-23 [有效至: 2017-05-17]
uid                  Wang Ziting <jysperm@gmail.com>
sig          C07CFB96 2016-05-04  paomian <qtang@leancloud.rocks>
sig 3        7CDC82A7 2015-05-11  Yeechan Lu <wz.bluesnow@gmail.com>
sig 3        E466CF1E 2016-05-17  Wang Ziting <jysperm@gmail.com>
sig          E411E711 2016-06-02  keybase.io/librazy <librazy@keybase.io>
sig          B0B002B8 2016-07-13  dennis (Dennis Zhuang) <killme2008@gmail.com>

uid                  keybase.io/jysperm <jysperm@keybase.io>
sig          C07CFB96 2016-05-04  paomian <qtang@leancloud.rocks>
sig 3        7CDC82A7 2015-05-11  Yeechan Lu <wz.bluesnow@gmail.com>
sig          E411E711 2016-06-02  keybase.io/librazy <librazy@keybase.io>
sig 3        E466CF1E 2016-05-17  Wang Ziting <jysperm@gmail.com>
sig          B0B002B8 2016-07-13  dennis (Dennis Zhuang) <killme2008@gmail.com>
```

GPG Public Key

- root public key：整个 GPG 账户的标识，RSA 的公钥（E466CF1E）
- uid：名字、邮件地址、头像（Wang Ziting <jysperm@gmail.com>）
- sub key：由 public key 授权的子密钥，可以为每个子密钥设置权限（SCEA）
  - 2048R/1D795875 created: 2014-11-23 expires: 2022-11-21 usage: S
- sig：对其他人的「信任签名」、他人对你的「信任签名」
- expired：可以为 sub key 或 sig 设置有效期
- revoke certificate：泄露 key 时用于通知其他人私钥已经泄露

gpg --sign

```
> echo 'hello' | gpg --clearsign -a
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

hello
-----BEGIN PGP SIGNATURE-----

iQIcBAEBAgAGBQJXfifAAAoJELQPOvDkZs8epZAP/3/jP6k1Dev2a8i8KfY7VDfv
TVGl61kLEbgpgR3mWXFL7PaJ8SyW8N0Dv3cJhYbY8NGp8wbkZa7cUS7DkTb2ArhS
M+IKUJtwUwbfp5fOyT+esaDLWatqjSJ+5IjWX8BOnh5SnLNMURDxsYrJMShfecTD
tbBfnEkIeCFBwIfE0Xs5m23+6i7t77ZgLdn1qpWLTRNpd6Fzi0B653Kr8dREPflI
MAsn6CP90pX55V0LnsZGiAgZV+34iFolhFDd7N5mtPT/zF7OToN2SNJF3YOVikBp
M+1WJL9W8x9DwzhOq8AmPgHIEwBZVNS8Nv+UNadIZJuexR0ERl64e8MdTLft0Qui
ChjUiD7ibkLR433jcms+2EJ04xd6Ie0mp/nH5nMLY1mEgHtLMXql6VHQCbJt80Vf
ZrL2J+BF9Sk1zPh9Hn5NGe+RLX1d/CZ62rYMICRcEwiS9vpWq6m9ouSMNZUYr8S5
a/ooD6gc71t457pVgkMqjo3Auazf4PRilUsAraQZilr+8yPhciE/PX3gBL5CtKHJ
4vKH9P9RngigL6D+YyBB5vMcpXlhx9ShbH2qLr106adJ1XrCpGtSmfxygjRn3xX9
Q1dWUaELUahhcdtK6IwZ6qzyp9AESpSd+Z/bnV7jc8iC0VtMOXipVnMo7J5qyHKD
/+yt8/tzoC4+0MEzlaHJ
=aOto
-----END PGP SIGNATURE-----
```

git tag -s v1.0.0

leanengine-node-sdk> git tag -v v1.1.0
object b0e42636df7394ac34f2b61c589233d0c3296d10
type commit
tag v1.1.0
tagger jysperm <jysperm@gmail.com> 1467084786 +0800

Release 1.1.0
gpg: 于 二 6/28 11:33:17 2016 CST 创建的签名，使用 RSA，钥匙号 E466CF1E
gpg: 完好的签名，来自于“Wang Ziting <jysperm@gmail.com>

Release

```
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA512

5c711a62e464f1d455c31afe2a62e9866eeaf1c23d977b57a60285d0bd040ba5  node-v6.3.0-darwin-x64.tar.gz
4f9a19b5688092d5652e8748093cb9c90fa503ea13b976e51dbd53c2fa07c116  node-v6.3.0-darwin-x64.tar.xz
9275894c3ed6373068cddeeb968e4d2ceba76368b6cd3b01aca79f0b592badd6  node-v6.3.0-headers.tar.gz
853589835867 60b0a9c0d36294ca5266dbd4e5d0df5390dfbf7aba684799e8db  node-v6.3.0-headers.tar.xz
58995c3f91962fc4383696f9c64763b3cd27d9b5903b4cf2a5ccfe86c8258e9f  node-v6.3.0-linux-arm64.tar.gz
18000ebe5208a2ddb17ab6d301a79d6ffa29287d579299480c62859c42c6c51c  node-v6.3.0-linux-arm64.tar.xz
de3554545e2d04719ebcd990984ff1eb5d6edbbbb9d24893cb998e2eb15d8bf5  node-v6.3.0-linux-armv7l.tar.gz
dce3e835d17a4febfb234f37593eb1b1dc02c87c16f49504c1800de7a8ccb0f2  node-v6.3.0-linux-armv7l.tar.xz
-----BEGIN PGP SIGNATURE-----
Comment: GPGTools - https://gpgtools.org

iQIcBAEBCgAGBQJXfUO5AAoJEEX17r2BPa6Ob0cQAIsV4/DC/961LkhIV2RJUXUO
YhFC6nchaTM3MSr3RQvDQQNC3TmEt9JagPPyMC1oMiy9DT7rzq52UCquIMh6nS8j
NjBs+L55Y0BaXrQbuqGs+isiqqUVvuLZkLU9IBbKs8hID4egqrHyY7tkxidoO3kL
g+z/A43rcSOPZzrGl6muwIe9OF/81o1sJHTETfn939SXfM9zOXQmbo71EfUnVNHT
ZfEIk4o4v24SR7+OW444ziJpJY4robJcFnUlA0WvBkf34ByAKIWnXm5yEEaNIIHa
R3E84JrXM0HgDfCVQ089pAyY1ImLT4XosK3j1pmjND/nOHHblH/TfcF73pBqtC70
n3eXjilVaTzZeMnRFHi5Ofopc9VPgAhAd/+Ya1aVDzUb0IA5mR3uMR8KTYx1UDZN
5yuBpHLhj4TFYmjHVvPRsJ4VT7qSUtHZXzHJyEG+mqhjJQtjkbF81fHbO0NYzr3G
uY0iUdc+cKmGuWq059qFJFo/6mlPdIlIydCIs31peVwRpI4JuXapGuVwRvlyD/kH
cFmElIXdq7S+Kn9jxIaokx2bo1hoZ1Q8Tb74s0kxhh+1bUlQohuUScvS3h1PG7cM
nn8OdMAVg9tNs2YrCndYzWqup80l+23boLTX9qyxEWeQo2Nb9Ddja2oZXERLy4hT
+v4vjY1hcMdVt/eL5tZ+
=DrfW
-----END PGP SIGNATURE-----
```

gpg --encrypt

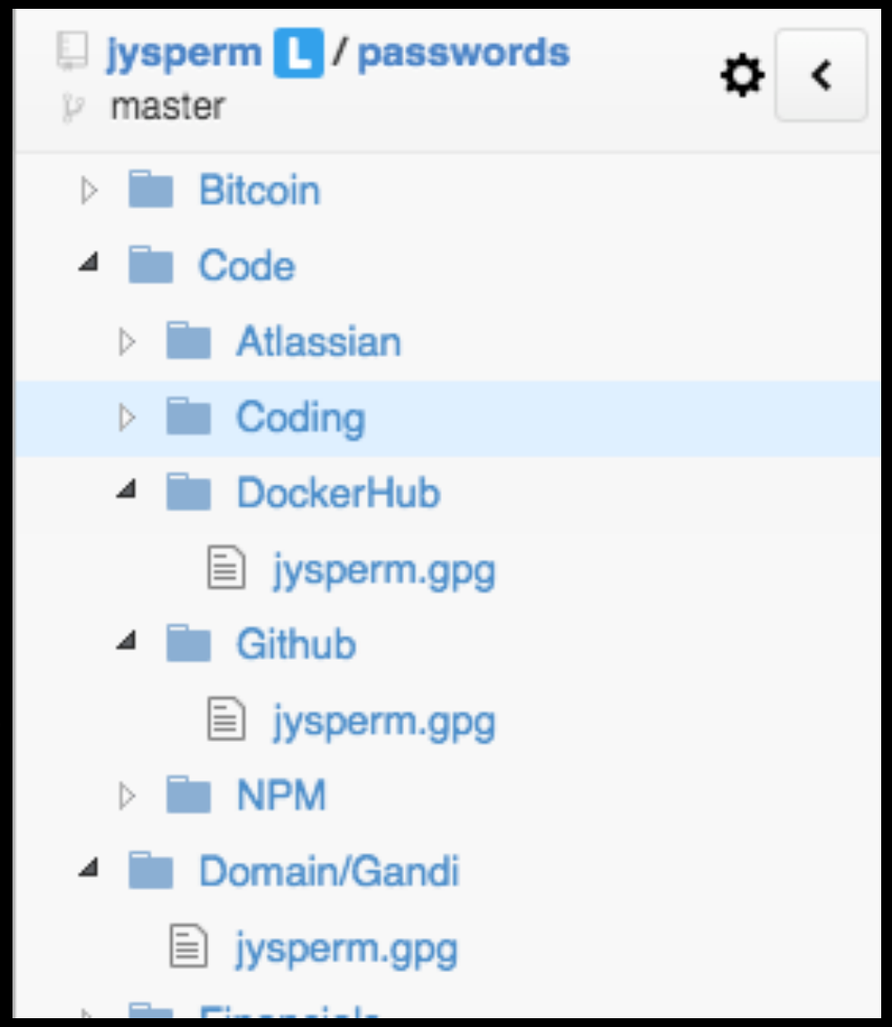
```
jysperm@jysperm-MacBook-Pro ~> echo 'hello' | gpg --encrypt -a -r jysperm
-----BEGIN PGP MESSAGE-----

hQEMAyp325QokoazAQf/RZACIGHK8dsgfwchf06/plemSlVHdYOmY4ipBuPDvui2
x/4HV15HC6i4qK4TkGF61lWElIj0BHikdREtPrqI2t4hsiRPZjVlX9Vic2BHLOnT
PErzRs5xUZcE5WQOFU4XmAqHhGwfmgw1SmS1N0411Hx2FakHgqI5kPvEePP1lkLT
o35tzJvnuuP5GXkLVhHAfHtMq0jDPSQEpvkn/k2tZeFgtI4mhwQrq1+dAGKStPIB
ptc2kTced1/pa8Jc7jAzdNY5oboluWq2rxyKXyLX0hsDfBt66c5a3v0X2mXR1kaW
l7V2Pw069+MMhqeTNq6hu9BXTWA6meWyUoGzSXp3KtI7AWj78aw6gajo1X7QMlcI
2vUGuh5CymMNrAHYNH95v31MU6lOJtArxiMTlDL58R5vwf7K9yTy2zXevbM=
=bZ2W
-----END PGP MESSAGE-----
```

```
jysperm@jysperm-MacBook-Pro ~> gpg -d
-----BEGIN PGP MESSAGE-----

hQEMAyp325QokoazAQf/RZACIGHK8dsgfwchf06/plemSlVHdYOmY4ipBuPDvui2
x/4HV15HC6i4qK4TkGF61lWElIj0BHikdREtPrqI2t4hsiRPZjVlX9Vic2BHLOnT
PErzRs5xUZcE5WQOFU4XmAqHhGwfmgw1SmS1N0411Hx2FakHgqI5kPvEePP1lkLT
o35tzJvnuuP5GXkLVhHAfHtMq0jDPSQEpvkn/k2tZeFgtI4mhwQrq1+dAGKStPIB
ptc2kTced1/pa8Jc7jAzdNY5oboluWq2rxyKXyLX0hsDfBt66c5a3v0X2mXR1kaW
l7V2Pw069+MMhqeTNq6hu9BXTWA6meWyUoGzSXp3KtI7AWj78aw6gajo1X7QMlcI
2vUGuh5CymMNrAHYNH95v31MU6lOJtArxiMTlDL58R5vwf7K9yTy2zXevbM=
=bZ2W
-----END PGP MESSAGE-----
gpg: 由 2048 位的 RSA 密钥加密，钥匙号为 289286B3、生成于 2014-11-23
      “Wang Ziting <jysperm@gmail.com>”
hello
```

pass-store




```
jysperm@jysperm-MacBook-Pro ~> pass find Coding
Search Terms: Coding
└── Code
    └── Coding
        └── jysperm.gpg

jysperm@jysperm-MacBook-Pro ~> pass insert Code/Coding/jysperm

jysperm@jysperm-MacBook-Pro ~> pass show Code/Coding/jysperm
DzizKKVIy22aHQwm

jysperm@jysperm-MacBook-Pro ~> pass git pull --rebase
remote: Counting objects: 11, done.
remote: Total 11 (delta 1), reused 1 (delta 1), pack-reused 10
Unpacking objects: 100% (11/11), done.
From github.com:jysperm/passwords
   5026f34..e94a70f  master     -> origin/master
First, rewinding head to replay your work on top of it...
Fast-forwarded master to e94a70f8b42af5e1c13dd69246b156bbcb24a94c.
```

SSH

```
~> export SSH_AUTH_SOCK=/Users/jysperm/.gnupg/S.gpg-agent.ssh

~> ssh-add -l
2048 SHA256:wO93TcTQHZtltKfvS0jewFh0CMj4No6xnTegtB8FN+k
```

```
~> ssh git@github.com
Hi jysperm! You've successfully authenticated, but GitHub does
not provide shell access.
Connection to github.com closed.
```

keybase.io

—— GitHub 之于 GPG 社区

Signed:
with love

11/23/14 8:01pm

Using:

PGP fingerprint B40F 3AF0 E466 CF1E

Payload:


```
{
  "body": {
    "key": {
      "fingerprint": "6d4092b2237ab5a89d122326b
      "host": "keybase.io",
      "key_id": "b40f3af0e466cf1e",
      "uid": "e2bb3652356cd8f11ae063f21514a000"
      "username": "jysperm"
    },
    "service": {
      "name": "twitter",
      "username": "jysperm"
    },
    "type": "web_service_binding",
    "version": 1
  },
  "ctime": 1416744085,
  "expire_in": 157680000,
  "prev": null,
  "seqno": 1,
```


Signed payload:
PGP fingerprint
B40F 3AF0 E466
CF1E + payload =


```
-----BEGIN PGP MESSAGE-----
Version: Keybase OpenPGP v1.1.6
Comment: https://keybase.io/crypto

yMGYAnicO8LLzMDFaC8m2yxbGVHKePrA6lKGkMIzU6uVkvJ
empRQVFmXomSlZJZiomBpVGSkZGxeWKSaaKFZYohkG1klmR
mKqko5SRXwzSATQmKbE4VS8zHygG5MRnpgBFsagvBUukGiU
GiamGpgZpxkZmhqaJBoYGIAUFqcW5SXmpgJVZ1UWF6QW5Sr
lpRnlpSkFuHUUVJZABIpT02Kh2qOT8rMSwH6F6ilLLWoODM
xNDM3MTEwMJURym1oiCzKDU+E6TC1NzMAug0Ax2lgqLUMiW
```


王子亭
@jysperm

关注

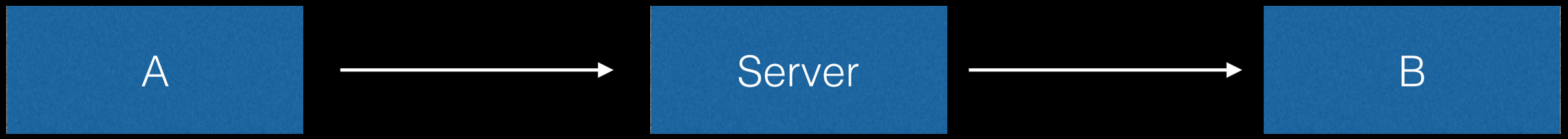
Verifying myself: I am jysperm on Keybase.io. pUWH16BH46vq1HWnRsjmmdxHQi5wlRxxglEf / keybase.io/jysperm/sigs/p …

上午4:01 - 2014年11月23日

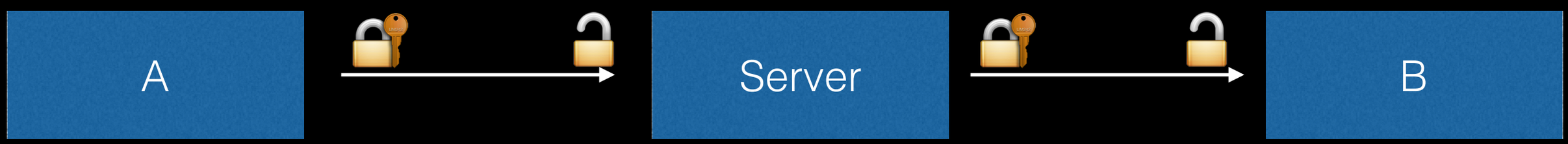
http://E466CF1E.pub

End-to-end Encryption

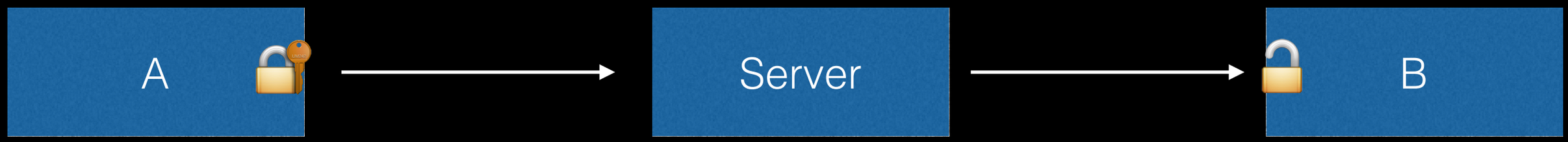
Without Encryption



SSL



End-to-end Encryption



E2EE in IM

nsun 这位兄台你钱包掉了 10:46 ☆

上传了文本 未命名-104648.txt

```
-----BEGIN PGP MESSAGE-----
Version: GnuPG v2

hQIMA84YxXdrpVuBARAAwm4CoVyYssn6A7IK3z9+INDZpe/IbOsIJ05WVrI4aP/a
bOfJtjC/Hm00Z55HdcYcUWEQl8YREL1g/JX8fkR+XM/mYT1ubrEpYerpNDT/QYJF
SNGq/SEdyEBdmQDkyGLAUDylhsSQFpgJ8Xe7GptTq/2W+OslwNo0vb98MgqF+toS
qvbYsQHTJNj7+PRMkYP4g6m8rHMErPd+JbYbYWnI2k2UV9VEDLLr/ZJBGrlRI5LL
Ddw7ZyvDebyXIW8nFubxHW7ADkzx+P6Xh907MUaaWrBtxS8U/gcZ2RADSce2Ggr2
HnUuWEc4wTE04n6BAv5FG9lJjNvYNyUb+GD4M9M5a09G2N3uKZ6xjcXTQ2kIbRhP
ov5p82ZzLLIrFCvW0Cw91sXq7upm9AUnqXLUzbjJ2PG1B56bOgqs3nA53wT39HoS
......
```

评论 (0) 下载 (1.6 KB) 转发 新窗口打开

qtang 东莞保健 10:56

-----BEGIN PGP MESSAGE-----

Version: GnuPG v2

hQEMA3nhMUlp/Pj6AQgAii3ugUeLRRP9X1RlSGNiESoKybCnmVmd3BjJO33UFUh5

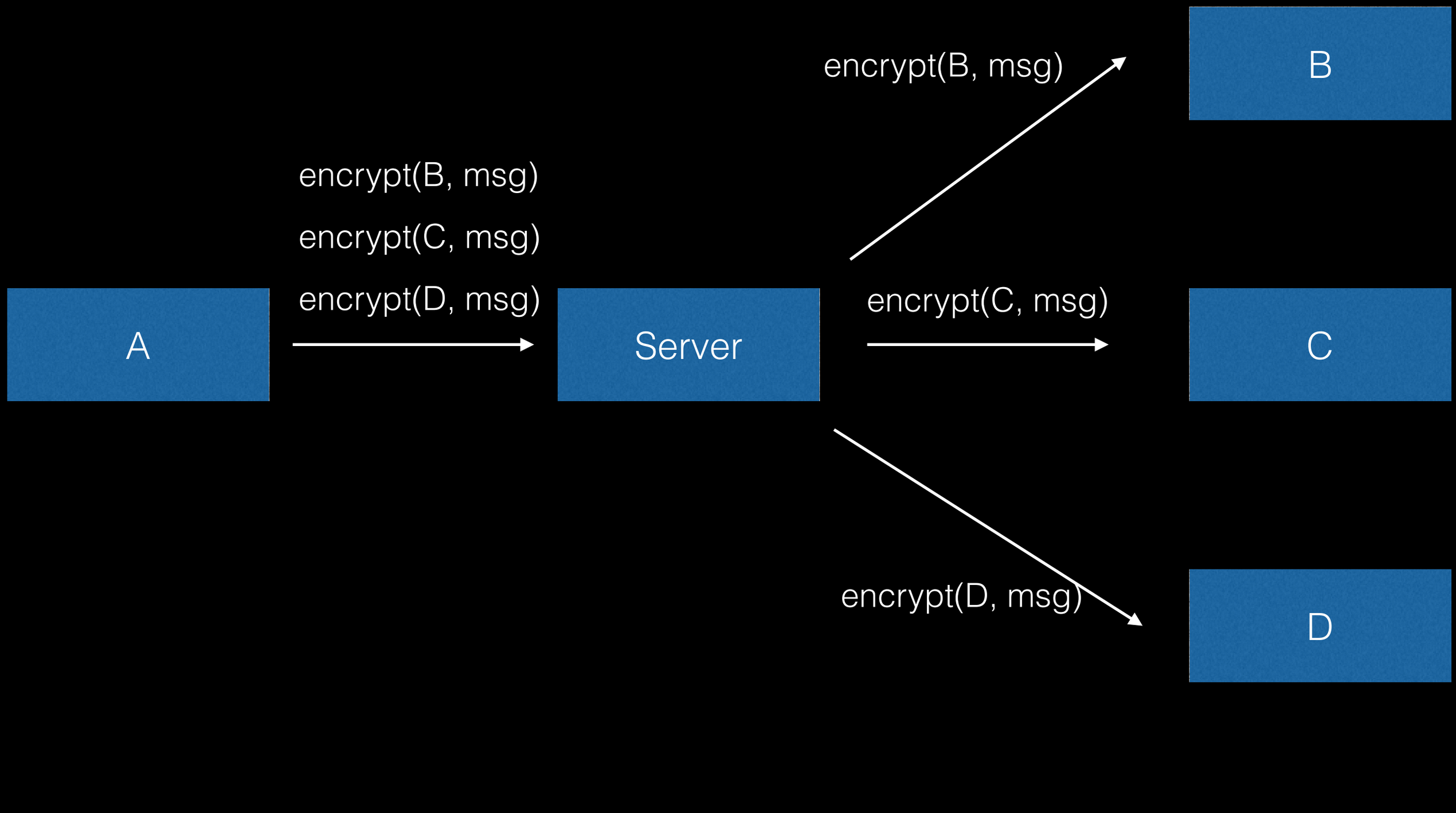
0WjamQvfGuSMW0NaFAPQd9UD1Mh+CViddfzwtWDFj3/v/zED+W8Jtnyy2kxUpmio

iMessage

- iMessage 会在每台设备上生成一个 RSA 密钥用于加密和一个 ECDSA 密钥用做签名。
- 公钥会被上传到 Apple IDS，其他用户会从 IDS 取得你的所有公钥（每个设备一个公钥）
- 发出的消息会用自己的 ECDSA 签名、用接收者的所有公钥分别加密一份，通过 APNs 发送给接收者。
- 多媒体消息使用临时密钥加密并通过 iCloud 传输，密钥则以加密的形式通过 APNs 发送。
- 包括 iCloud、Facetime、Keychain 均使用类似的端到端加密技术。

https://www.apple.com/cn/business/docs/iOS_Security_Guide.pdf

Group Chat


encrypt(B, msg)
B
encrypt(B, msg)
encrypt(C, msg)
A
encrypt(D, msg)
Server
encrypt(C, msg)
C
encrypt(D, msg)
D

防止密钥丢失

备份、Secret sharing

```
% ssss-split -t 3 -n 5
Generating shares using a (3,5) scheme with dynamic security
level.
Enter the secret, at most 128 ASCII characters: my secret root
password
Using a 184 bit security level.
1-1c41ef496eccfbeba439714085df8437236298da8dd824
2-fbc74a03a50e14ab406c225afb5f45c40ae11976d2b665
3-fa1c3a9c6df8af0779c36de6c33f6e36e989d0e0b91309
4-468de7d6eb36674c9cf008c8e8fc8c566537ad6301eb9e
5-4756974923c0dce0a55f4774d09ca7a4865f64f56a4ee0

% ssss-combine -t 3
Enter 3 shares separated by newlines:
Share [1/3]: 3-fa1c3a9c6df8af0779c36de6c33f6e36e989d0e0b91309
Share [2/3]: 5-4756974923c0dce0a55f4774d09ca7a4865f64f56a4ee0
Share [3/3]: 2-fbc74a03a50e14ab406c225afb5f45c40ae11976d2b665
Resulting secret: my secret root password
```

防止密钥泄露

「存在硬盘上的私钥早晚要泄露」

– Wang Ziting

硬件密钥

- Yubikey（OpenPGP SmartCard）
- iPhone
- TPM

小结

- GPG 可以在互联网上，从数学的角度为你创造一个无法被伪造的身份，并以此身份签名信息、接收加密信息。
- GPG 使用去中心化的模型，需要自行通过多种渠道来交换公钥，因此不会受制于单一的权威机构。
- GPG 提供了身份管理和相互进行「信任签名」的机制来简化密钥的交换过程。
- GPG 是一个开放的标准（兼容很多软件和硬件），有着活跃的社区，提供了相对易用的工具来进行公钥加密、解密、签名、验签。
- 基于公钥加密并签名的端到端加密是从理论上保证通讯安全的唯一方法，但在此之前你需要通过某种方式来交换公钥。
- 如果私钥丢失就只能改头换面、重新做人了。