

Jabber security

Peter Saint-Andre

stpeter@jabber.org



<https://stpeter.im/>

secure communication

with Jabber

what is Jabber?

open XML technologies

real-time messaging

presence

multimedia negotiation

collaboration

and more

invented by Jeremie
Miller in 1998

in essence...

streaming XML

over long-lived
TCP connection

client-server architecture

**decentralized
network**

inter-domain messaging

like email

but really fast

with built-in presence

not just one
open-source project

multiple codebases

**open-source and
commercial**

interoperability via
XML wire protocol

core protocol
standardized @ IETF

Extensible

Messaging

and

Presence

Protocol

(XMPP)

RFCs 3920 + 3921

multiple
implementations

serious deployment

how many users?

we don't know

**decentralized
architecture**

100k+ servers

50+ million IM users

not just IM

generic XML routing

lots of applications
beyond IM

continually defining
XMPP extensions

XMPP Standards Foundation (XSF)

developer-driven
standards group

that's great, but...

how secure is it?

what is security?

**secure conversation
in real life...**

a good friend
visits your home

you know and trust
each other

only the two of you

strangers can't enter
your home

**your home is not
bugged**

**conversation is not
recorded**

what you say is private
and confidential

**contrast that with
the Internet...**

lots of potential attacks

man-in-the-middle

eavesdropping

unauthenticated users

address spoofing

weak identity

rogue servers

denial of service

directory harvesting

buffer overflows

spam

spim

spit

splogs

viruses

worms

trojan horses

malware

phishing

pharming

information leaks

**inappropriate logging
and archiving**

you get the picture

**the Internet is a
dangerous place**

how do we fight
these threats?

sorry, but...

**Jabber is not a
perfect technology**

**not originally built
for high security**

don't require PGP keys
or X.509 certs

**don't require ubiquitous
encryption**

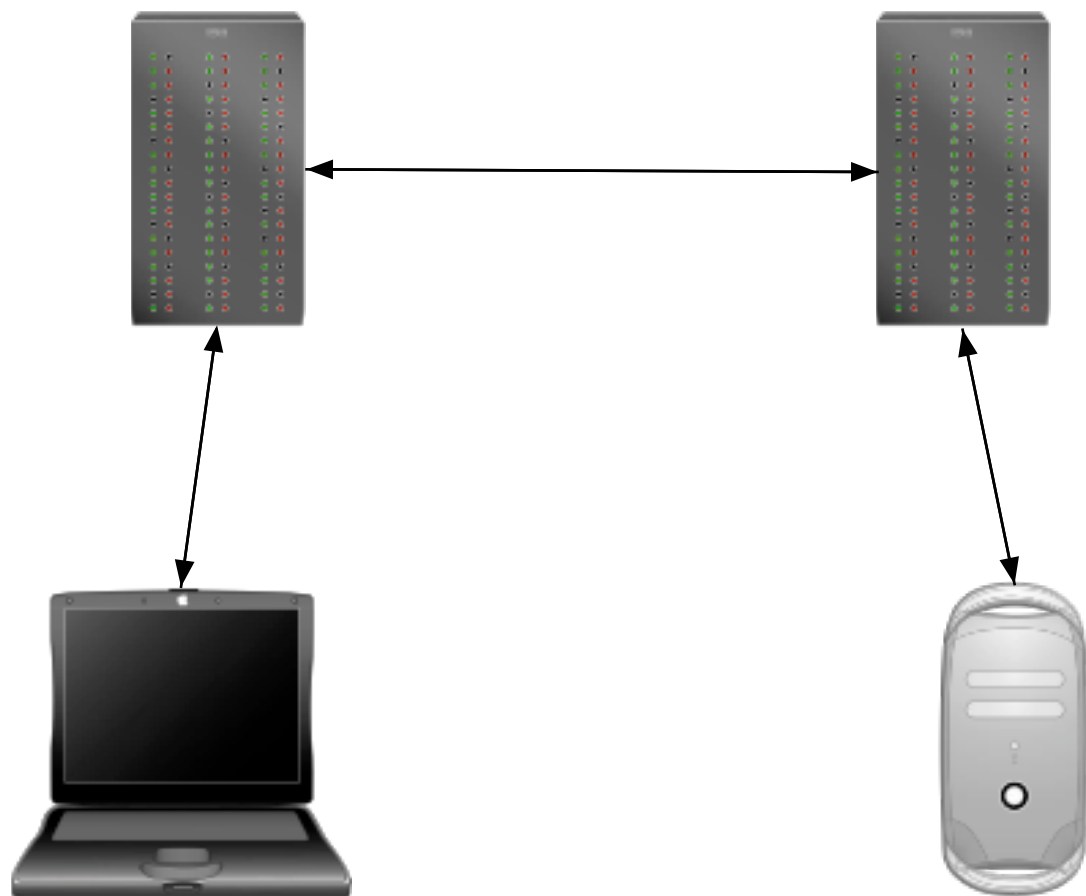
tradeoffs between
security and usability

maybe that's why we
have 50+ million users...

but privacy and security
are important

so what have we
done to help?

Jabber architecture...



similar to email

**client connects to
server (TCP 5222)**

(or connect via HTTP
binding over SSL)

**client MUST
authenticate**

originally: plaintext (!) or
hashed password

Simple Authentication & Security Layer (SASL)

RFC 4422

many SASL mechanisms

**PLAIN (OK over
encrypted connection)**

DIGEST-MD5

**EXTERNAL (with
X.509 certs)**

GSSAPI (a.k.a. Kerberos)

ANONYMOUS

etc.

**all users are
authenticated**

sender addresses not
merely asserted

**server stamps
user 'from' address**

**Jabber IDs are
logical addresses**

**IP addresses
not exposed**

**Jabber ID looks like
an email address**

romeo@montague.net

juliet@capulet.com

**not limited to
US-ASCII characters**

jiří@čechy.cz

πλατω@ελλάς.gr

มณำปจ@jabber.th

ぷおぞ@jabber.jp

∞ @math.it

full Unicode opens
phishing attacks

STPETER@jabber.org
STPETER@jabber.org

clients should use
“petnames”

store in buddy list [tm]
(a.k.a. “roster”)

**server stores
your roster**

**server broadcasts
your presence**

**but only to subscribers
you have authorized**

most traffic goes
through server

traffic is pure XML

**servers reject
malformed XML**

**servers may validate
traffic against schemas**

difficult to inject
binary objects

difficult to propagate
malware

**break alliance between
viruses and spam**

spam virtually unknown
on Jabber network

why?

**hard to spoof
addresses**

hard to send
inline binary

**XHTML subset
(no scripts etc.)**

**user approval required
for file transfer**

privacy lists to block
unwanted users

**XMPP not immune
to spam**

have spam-fighting tools
ready when it appears

**challenge-response to
register an account**

**challenge-response to
communicate**

spam reporting

working on more
anti-spam tools

**server reputation
system?**

**anonymized IP address?
(groupchat spam)**

**spammers need to
overcome...**

bandwidth limits

connection limits

**other denial-of-service
prevention measures**

**distributed attack or
run a rogue server**

not impossible

just harder than other
networks (got email?)

**no rogue servers
(yet)**

optional to federate
with other servers

many private
XMPP servers

**public servers federate
as needed (TCP 5269)**

**DNS lookup to get
server IP address**

**only one hop
between servers**

**server identities
are validated**

**server dialback
(identity verification)**

**effectively prevents
server spoofing**

receiving server checks
sending domain

no messages from
“service@paypal.com”

**DNS poisoning
can invalidate**

**need something
stronger?**

Transport Layer Security (TLS)

RFC 4346

IETF “upgrade” to SSL

TLS + SASL EXTERNAL
with X.509 certs

**strong authentication
of other servers**

but only if certs are
not self-signed

\$\$\$

real X.509 certs
are expensive

VeriSign, Thawte, etc.

a better way...

xmpp.net

intermediate CA for
XMPP network

**free digital certificates
for XMPP server admins**

(need to prove you
own the domain)

root CA: StartCom

ICA: XMPP Standards Foundation

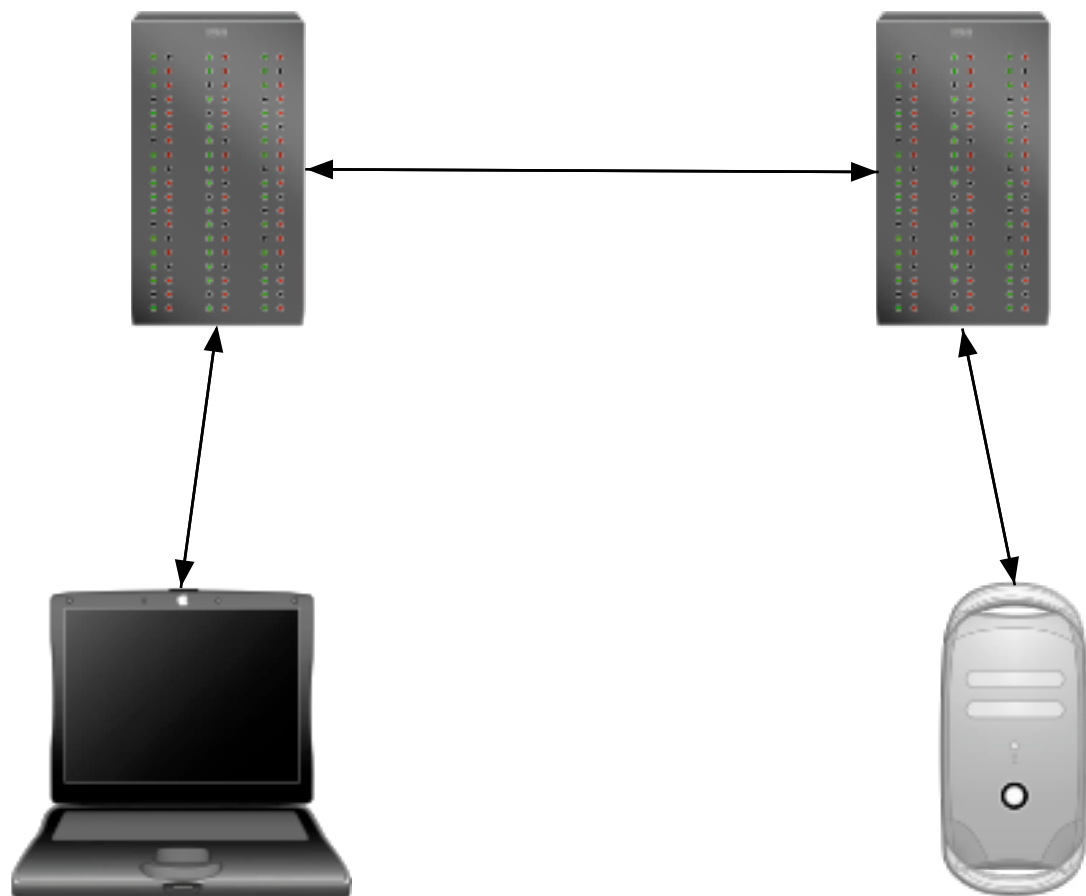
hopefully other CAs
in future

**so channel encryption
is a no-brainer**

**man-in-the-middle
is much harder**

“Mallory” is foiled

but what about
“Isaac” and “Justin”?



**we can encrypt
the channels**

**but traffic is cleartext
within servers!**

need end-to-end
encryption (“e2e”)

**first try: OpenPGP
(XEP-0027)**

great for geeks

**but Aunt Tillie
doesn't use PGP**

**second try: S/MIME
(RFC 3923)**

great for geeks (and
some employees)

**but Aunt Tillie
doesn't use X.509**

**XML encryption and
digital signatures?**

seems natural, but not
much interest (c | 4n?)

doesn't provide perfect
forward secrecy

**off-the-record
communication (OTR)?**

great idea

**opportunistic
encryption (à la SSH)**

perfect forward secrecy

**but encrypts only the
plaintext message body**

we need to encrypt
the entire packet

why?

because XMPP is more
than just IM

protect IPs sent in
multimedia negotiation

**protect shared XML
editing data**

etc.

**solution: encrypted
sessions**

**big set of
requirements...**

packets are confidential

packet integrity

replay protection

key compromise does
not reveal past comms

don't depend on public
key infrastructure

**entities authenticated
to each other**

3rd parties cannot
identify entities

**robustness against
attack (multiple hurdles)**

upgradeability if bugs
are discovered

encryption of full
XMPP packets

implementable by
typical developer

usable by
typical user

**how to address all
requirements?**

just a dream?

bootstrap from
cleartext to encryption

in-band Diffie-Hellman key exchange

**translate “SIGMA”
approach to XMPP**

similar to Internet Key
Exchange (IKE)

**details in XSF XEPs
116, 188, 200**

**simplified profile
in XEP-0217**

major priority for
2007-2008

support from NLnet
(thanks!)

**pursuing full
security analysis**

code bounties

GSoC project

Jabber security summit

more at
blog.xmpp.org

wide implementation
in next ~12 months

so how are we doing?

spam free

hard to spoof addresses

pure XML discourages
binary malware

**DoS attacks possible
but not easy**

widespread channel
encryption

working hard on
end-to-end encryption

widely deployed in high-
security environments

Wall Street investment banks

U.S. military

**MIT and other
universities**

**many public servers
since 1999**

**no major security
breaches**

can't be complacent

always more to do

**security is a never-
ending process**

analysis and hacking
are encouraged

**if it breaks,
we'll fix it**

security@xmpp.org

join the conversation

**let's build
a more secure Internet**