Introduction to Public Key Cryptography

Email:

micahflee@riseup.net

micah@eff.org

5C17 6163 61BD 9F92
422A C08B B4D2 5A1E
9999 9697 *(GnuPG)*

Jabber:

micah@jabber.ccc.de

F38D9B47 35BD9AC1
3A5AEE1B AA42A761
1B2814E6 *(OTR)*

Twitter: @micahflee

Security Autonomy

- **Crypto is hard:** Even people who use it all the time don't understand why they have to do what experts tell them
- Anyone can follow instructions **without understanding** what's going on
- If you **understand the nuts and bolts**, you don't need training for specific software

Scope

This workshop covers:

- Secret keys, public keys, signatures
- Attacks against encryption and how to protect yourself
- Encrypting messages (IM, SMS, email)
- PKI, like HTTPS

This workshop does not cover:

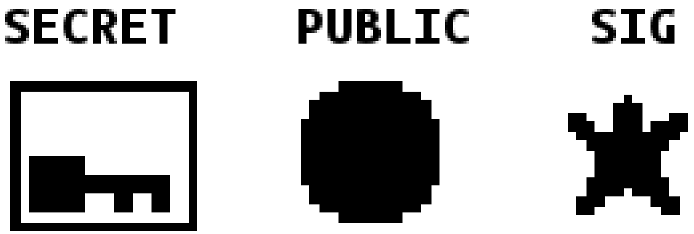
- The mathematical magic behind encryption
- How to use specific software

Props and Volunteers

- I brought a bunch of **props** to explain concepts
- I will need **volunteers**
- *You are my guinea pigs! There's a lot of stuff crammed in here. I'll try to get through as much as possible.*
- **I want you to understand these concepts, so please ask questions at any time**

Keys

- Each person who uses public key crypto has a **key pair**, public key and secret key
- **Secret key**
  - Keep it secret, keep it safe
  - Linked to one's identity
- **Public key**
  - Publish wide and far
  - Lots of copies



Alice, Bob, and Eve

- **Alice** and **Bob** are just two folks trying to communicate
- **Eve** is the eavesdropper
  - She can monitor and modify all messages
  - Maybe she works at your ISP, or works with the NSA, or hangs out at the same coffee shop as you

Sending a Message: Eavesdropping and Modifying

Description:

- Alice sends message to Bob
- Eve eavesdrops
- Bob replies to Alice
- Eve modifies
- Eve sends message to Bob

Parties:

- Alice
- Bob
- Eve

SECRET	PUBLIC	SIG
▣	●	★

Encrypting and Decrypting

Description:

- Alice asks for Bob's public key
- Bob gives copy of public key to Alice
- Alice encrypts message using Bob's public key
- Alice sends ciphertext to Bob
- Bob uses secret key to decrypt ciphertext
- Passive Eve can't eavesdrop

Parties:

- Alice
- Bob
- Eve

SECRET	PUBLIC	SIG
🔑	●	★

Imposters

Description:

- Eve pretends to be Alice, asks for Bob's public key
- Eve encrypts message to Bob, signs Alice's name
- Bob decrypts, has been duped

Parties:

- Bob
- Eve

SECRET	PUBLIC	SIG
⚿	●	★

Spies: Part 1

Description:

- Alice asks for Bob's public key
- Eve intercepts public key!
- Eve gives Alice her own public key
- Alice encrypts message to Eve's public key, thinking it's Bob

Parties:

- Alice
- Bob
- Eve

SECRET	PUBLIC	SIG
🔑	●	★

Spies: Part 2

Description:

- Alice sends message to Bob, but Eve intercepts!
- Eve decrypts message using her secret key
- Eve re-encrypts message to Bob's public key
- Eve sends message to Bob

Parties:

- Alice
- Bob
- Eve

SECRET	PUBLIC	SIG
🔑	●	★

Spies: Part 3
What just happened?

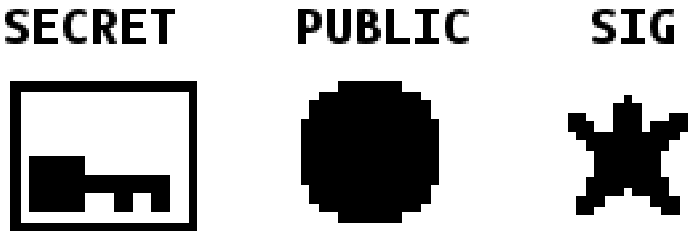
- That was a man in the middle (woman in the way?) attack
- If you are chatting with someone and your conversation is unverified, **you have no way of knowing if this is happening to you**
- You can be talking to your **real friend**, **encrypting everything**, with your **enemy listening in**

Spies: Part 4
What just happened?

- This might happen to you if you are using:
  - Pidgin/Adium and OTR
  - TextSecure on Android
  - PGP/GnuPG
  - SSL-enabled internet service: HTTPS, SSH, etc.
- Verifying identity solves this problem

Signatures

- You can use your secret key to **digitally sign** something
- Other people who have your **public key** can **verify your signature**
- It is **impossible to fake** a digital signature (with some exceptions :p)



Signing Messages
To Prevent Tampering

Description:

- Alice writes a message and signs it
- Alice sends it to Bob
- Eve intercepts! Modifies the message but leaves the same signature
- Bob sees the message is signed with an invalid signature

Parties:

- Alice
- Bob
- Eve

SECRET	PUBLIC	SIG
🔑	●	★

Signing Messages
Tampered Anyway

Description:

- Alice writes a message and signs it
- Alice sends it to Bob
- Eve intercepts! Modifies the message, and signs it herself
- Bob sees the message is signed with a valid signature

Parties:

- Alice
- Bob
- Eve

SECRET 🔑 **PUBLIC** ● **SIG** ★

Signing Messages
Tampered Anyway (cont.)

- Just because a message is **from your friend** and is **digitally signed** (but unverified) doesn't mean **your enemy** didn't sign it!
- **Solution is for Alice and Bob to confirm each other's public keys**

Signing Keys

Description:

- Alice and Bob meet **in person** at a **CryptoParty** (or maybe they talk on the phone)
- Alice gives Bob a copy of her public key, and **Bob signs it**
- Bob gives Alice a copy of his public key, and **Alice signs it**

Parties:

- Alice
- Bob

SECRET 🔑 **PUBLIC** ● **SIG** ★

Signing Messages
Eve Gets Caught Tampering

Description:

- Alice writes a message and signs it
- Alice sends it to Bob
- Eve intercepts! Modifies the message, and signs it herself
- Bob sees the message is signed with a valid signature, **but not Alice's!**

Parties:

- Alice
- Bob
- Eve

SECRET	PUBLIC	SIG
🔑	●	★

Encrypting and Signing Messages

Description:

- Bob doesn't need to ask for Alice's key, he already has a copy he signed
- Bob writes a message, encrypts with Alice's public key, signs with his own key
- Bob sends message to Alice
- Eve sulks

Parties:

- Alice
- Bob
- Eve

SECRET	PUBLIC	SIG
🔑	●	★

Eve's Final Trick

Description:

- Eve writes a message to Bob, pretending to be Alice
- Eve encrypts it to Bob's public key
- Eve sends it to Bob
- Bob receives encrypted message “from Alice”, that isn't signed
- He decrypts it, reads it, but wonders why Alice didn't sign it
- Bob could just trust it, but instead he calls Alice on the phone to verify. **Eve gets caught!**

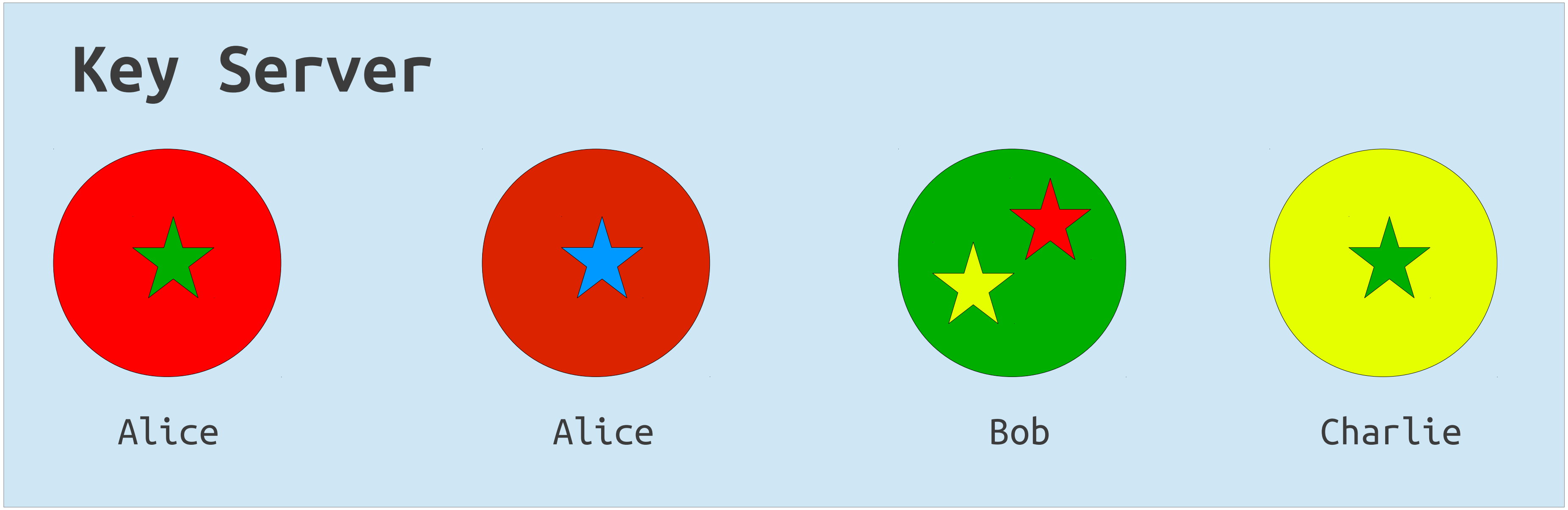
Parties:

- Alice
- Bob
- Eve

SECRET	PUBLIC	SIG
🔑	●	★

Thank you Alice and Bob!
(I'll need new volunteers later)

Web of Trust



Key Server

Alice

Alice

Bob

Charlie

Alice has signed Bob's key

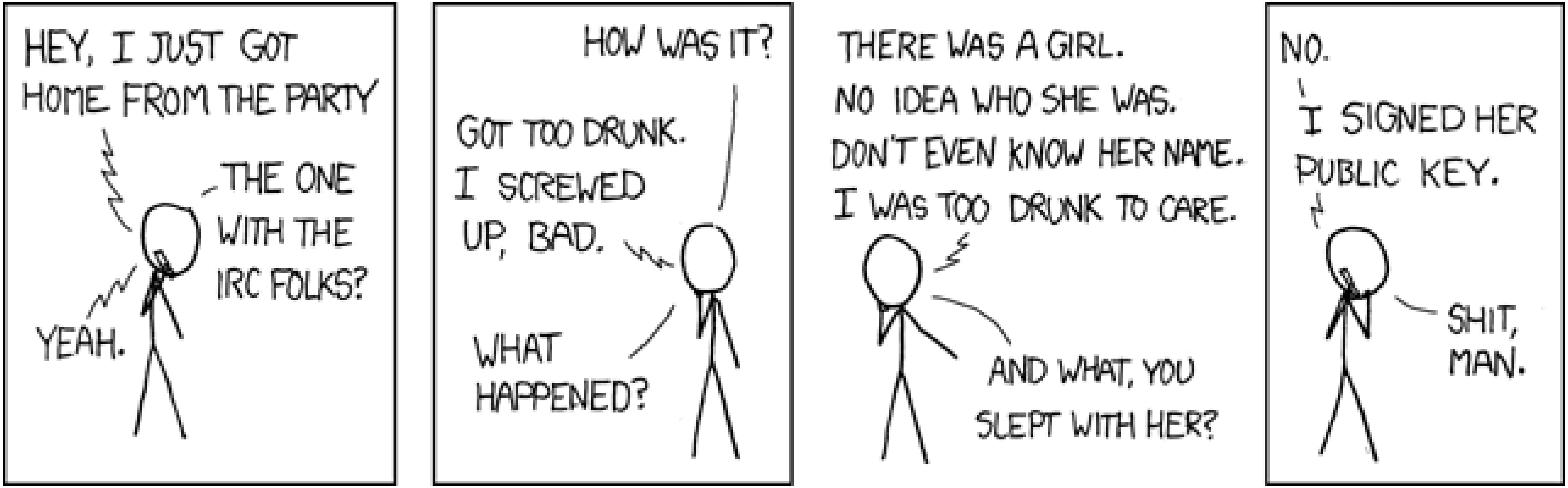
Bob has signed Alice's key

Charlie has signed Bob's key

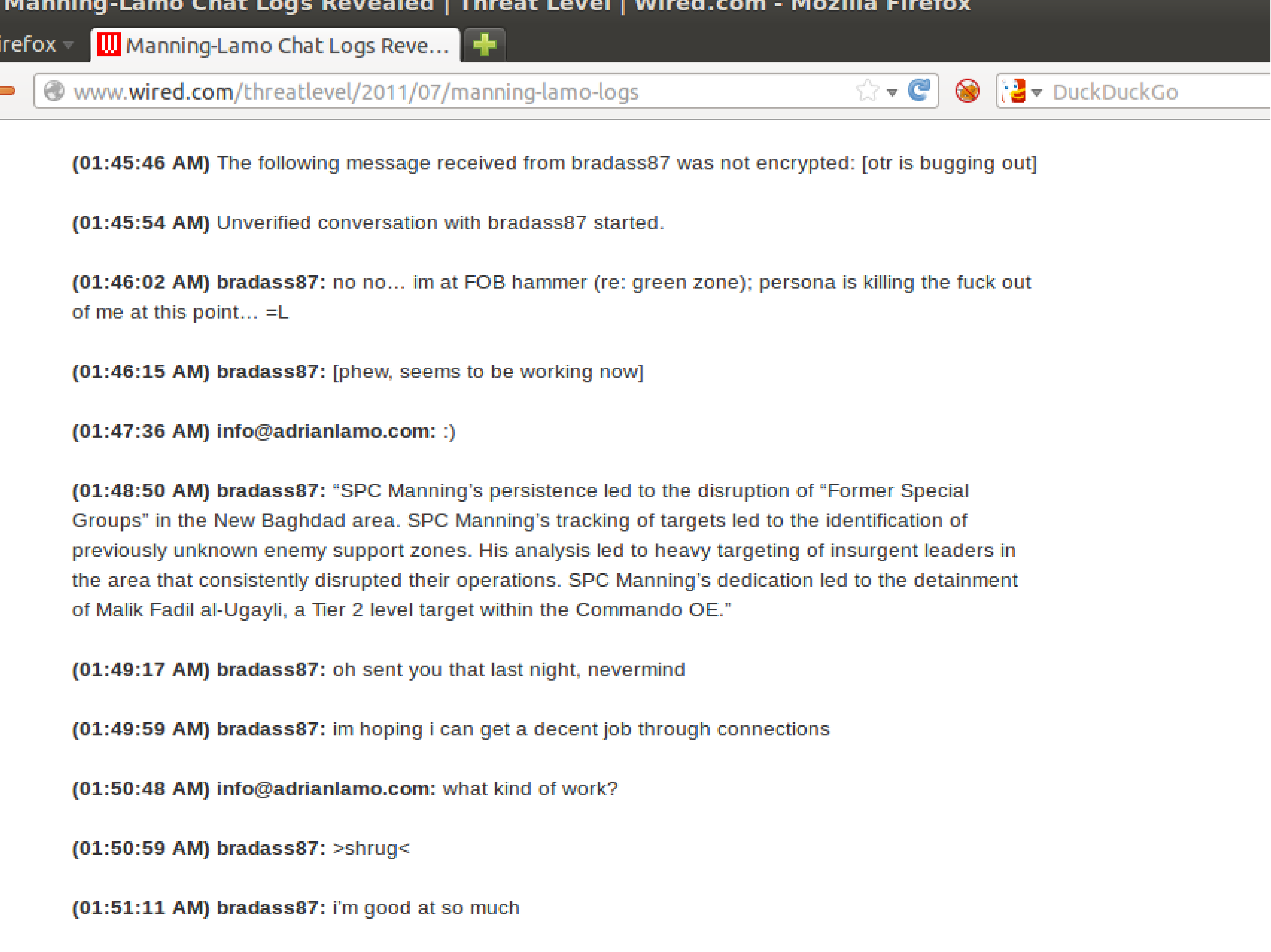
Bob has signed Charlie's key

Charlie needs to talk to Alice

Please Sign Responsibly


HEY, I JUST GOT HOME FROM THE PARTY
YEAH.
THE ONE WITH THE IRC FOLKS?
HOW WAS IT?
GOT TOO DRUNK. I SCREWED UP, BAD.
WHAT HAPPENED?
THERE WAS A GIRL. NO IDEA WHO SHE WAS. DON'T EVEN KNOW HER NAME. I WAS TOO DRUNK TO CARE.
AND WHAT, YOU SLEPT WITH HER?
NO.
I SIGNED HER PUBLIC KEY.
SHIT, MAN.

Thanks, XKCD!


Manning-Lamo Chat Logs Revealed | Threat Level | Wired.com - Mozilla Firefox
irefox
Manning-Lamo Chat Logs Reve…
www.wired.com/threatlevel/2011/07/manning-lamo-logs
DuckDuckGo

(01:45:46 AM) The following message received from bradass87 was not encrypted: [otr is bugging out]

(01:45:54 AM) Unverified conversation with bradass87 started.

(01:46:02 AM) bradass87: no no… im at FOB hammer (re: green zone); persona is killing the fuck out of me at this point… =L

(01:46:15 AM) bradass87: [phew, seems to be working now]

(01:47:36 AM) info@adrianlamo.com: :)

(01:48:50 AM) bradass87: “SPC Manning’s persistence led to the disruption of “Former Special Groups” in the New Baghdad area. SPC Manning’s tracking of targets led to the identification of previously unknown enemy support zones. His analysis led to heavy targeting of insurgent leaders in the area that consistently disrupted their operations. SPC Manning’s dedication led to the detainment of Malik Fadil al-Ugayli, a Tier 2 level target within the Commando OE.”

(01:49:17 AM) bradass87: oh sent you that last night, nevermind

(01:49:59 AM) bradass87: im hoping i can get a decent job through connections

(01:50:48 AM) info@adrianlamo.com: what kind of work?

(01:50:59 AM) bradass87: >shrug<

(01:51:11 AM) bradass87: i’m good at so much

HTTPS
Self-Signed Certificate

Description:

- Firefox is packaged with CA's public key
- Firefox tries to load Bank's website
- Bank gives Firefox it's public key
- Firefox sees that Bank's key is **not signed by CA**, throws scary warning

Parties:

- Certificate Authority
- Bank
- Firefox

SECRET	PUBLIC	SIG
🔑	●	★

HTTPS
CA-Signed Certificate

Description:

- Firefox is packaged with CA's public key
- Firefox tries to load Bank's website
- Bank gives Firefox it's public key **that's signed by CA**
- Firefox sees that Bank's public key is signed by CA, starts encrypted session

Parties:

- Certificate Authority
- Bank
- Firefox

SECRET	PUBLIC	SIG
🔑	●	★

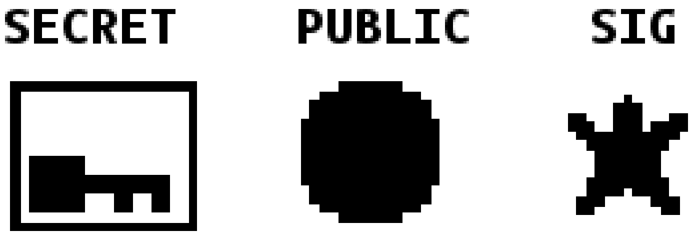
HTTPS
Man in the Middle

Description:

- Firefox tries to load Bank's website
- Eve intercepts! Eve tries to load Bank's website
- Bank gives Eve it's public key **that's signed by CA**
- Eve gives Firefox Eve's public key
- Firefox sees that the public key (Eve's) is **not signed by CA**, throws scary warning
- Eve sulks

Parties:

- Certificate Authority
- Bank
- Firefox
- Eve



HTTPS
CA-Signed Man in the Middle

Description:

- Firefox tries to load Bank's website
- Eve intercepts! Eve tries to load Bank's website
- Bank gives Eve it's public key **that's signed by CA**
- **Eve works for/owns/hacks CA, and signs her own public key**
- Eve gives Firefox Eve's public key **that's signed by CA**
- Firefox sees that the public key (Eve's) is **signed by CA**, starts encrypted session with Firefox
- **Everyone loses :(**

Parties:

- Certificate Authority
- Bank
- Firefox
- Eve

SECRET	PUBLIC	SIG
🔑	●	★

Thank you Certificate Authority,
Bank, and Firefox!

Certificate Authorities

- **CAs are a bit more complicated than this**
- Technically web servers use **certificates**, not public keys
- Browsers trust ~100 root CAs
- **Like vampires, CAs can sire new CAs** creating **intermediate CAs**
- New CAs are created when an existing CA digitally signs the “signing certificate” of a new CA
- If a web servers' CA is signed by an intermediate, the web server should serve the entire certificate chain, details details, blah blah blah...

Certificate Authorities (cont.)

- All 100 root CAs, plus all the intermediate CAs, adds up to **roughly 650 different organizations** (see: https://www.eff.org/observatory)
- If any one of them has **a malicious employee**, **gets hacked**, or **gets compelled by their government**, it can be used to man in the middle **any HTTPS website on the web**
- **The certificate authority system is broken**, but decentralized solutions are in the works:
  - Sovereign Keys: https://www.eff.org/sovereign-keys
  - Convergence: http://convergence.io/

Final Thoughts

- Encryption keys are just huge numbers, stored in a file on your hard drive
- You can backup your keys by backing up the right files on your hard drive

Final Thoughts

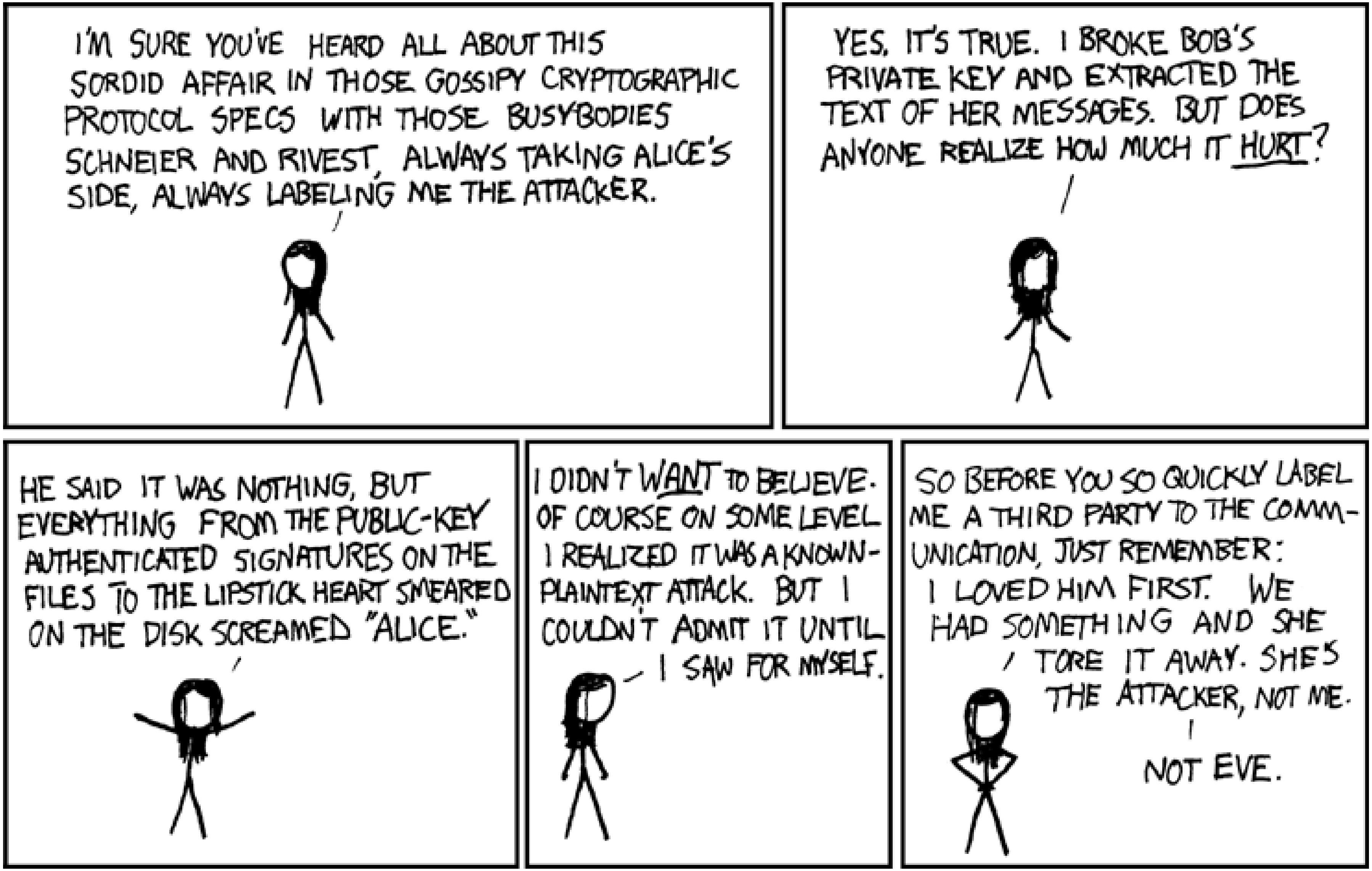
- When you use GPG or TextSecure, **your secret key is stored encrypted**
- When you type your passphrase, you are decrypting your secret key
- If you lose your phone or computer, your GPG and TextSecure keys are safe **as long as your passphrase is good**

Final Thoughts

- OTR (as implemented in Pidgin, Adium, Gibberbot, ChatSecure) **stores your secret key in plaintext**
- You don't have to constantly type an annoying passphrase to use OTR, which is convenient
- If your computer or phone is lost, **your secret key has been compromised**

Thank you! I would love to sign your key.

Email: micahflee@riseup.net 5C17 6163 61BD 9F92 422A C08B B4D2 5A1E 9999 9697
Jabber: micah@jabber.ccc.de F38D9B47 35BD9AC1 3A5AEE1B AA42A761 1B2814E6
Twitter: @micahflee


I'M SURE YOU'VE HEARD ALL ABOUT THIS SORDID AFFAIR IN THOSE GOSSIPY CRYPTOGRAPHIC PROTOCOL SPECS WITH THOSE BUSYBODIES SCHNEIER AND RIVEST, ALWAYS TAKING ALICE'S SIDE, ALWAYS LABELING ME THE ATTACKER.
YES, IT'S TRUE. I BROKE BOB'S PRIVATE KEY AND EXTRACTED THE TEXT OF HER MESSAGES. BUT DOES ANYONE REALIZE HOW MUCH IT HURT?
HE SAID IT WAS NOTHING, BUT EVERYTHING FROM THE PUBLIC-KEY AUTHENTICATED SIGNATURES ON THE FILES TO THE LIPSTICK HEART SMEARED ON THE DISK SCREAMED "ALICE."
I DIDN'T WANT TO BELIEVE. OF COURSE ON SOME LEVEL I REALIZED IT WAS A KNOWN-PLAINTEXT ATTACK. BUT I COULDN'T ADMIT IT UNTIL I SAW FOR MYSELF.
SO BEFORE YOU SO QUICKLY LABEL ME A THIRD PARTY TO THE COMMUNICATION, JUST REMEMBER: I LOVED HIM FIRST. WE HAD SOMETHING AND SHE TORE IT AWAY. SHE'S THE ATTACKER, NOT ME.
NOT EVE.