HEARTBLEED && WIRELESS

New attack vectors for heartbleed:

Enterprise wireless (and wired) networks

Luis Grangeia

lgrangeia@sysvalue.com | twitter.com/lgrangeia

28 / 05 / 2014 @ Confraria IT Security - Lisbon

WHAT IS HEARTBLEED

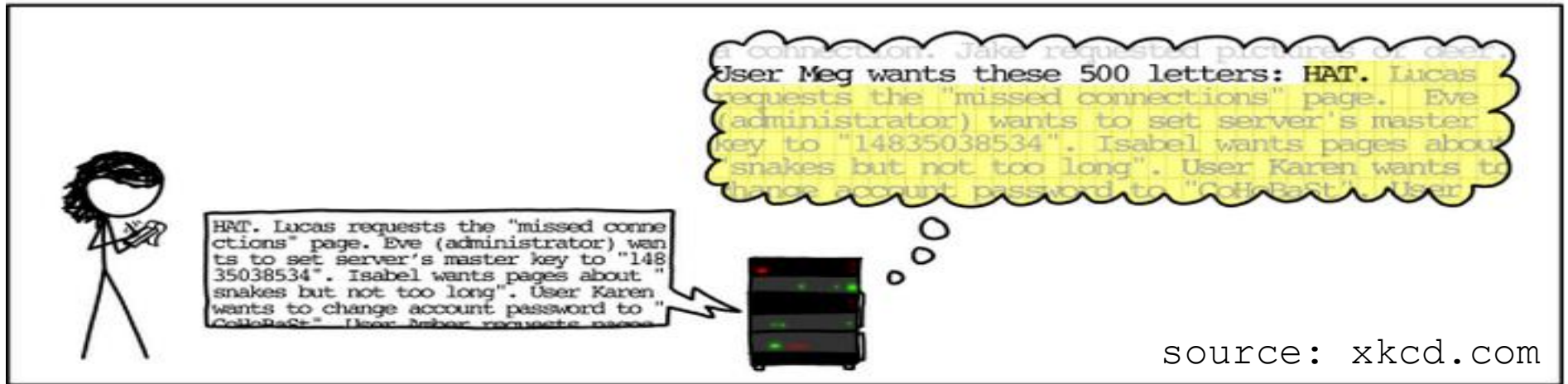
“Catastrophic bug” on OpenSSL:

"The Heartbleed bug allows anyone on the Internet to read the memory of the systems protected by the vulnerable versions of the OpenSSL software. This compromises the secret keys used to identify the service providers and to encrypt the traffic, the names and passwords of the users and the actual content. This allows attackers to eavesdrop communications, steal data directly from the services and users and to impersonate services and users.

“On the scale of 1 to 10, this is an 11.” - Schneier

HOW THE HEARTBLEED BUG WORKS:

source: xkcd.com



source: xkcd.com

WHAT IS “ENTERPRISE WIRELESS”

- WPA / WPA2 Networks
- Protected by multiuser authentication
- Tipically using a EAP Method:
  - **EAP-PEAP**
  - **EAP-TTLS / EAP-TLS**
  - EAP-SIM / EAP-AKA

EAP AND TLS

- EAP-PEAP, EAP-TTLS, EAP-TLS
- All these use a **TLS tunnel** over EAP to secure some part of the authentication process
- EAP... OpenSSL... Heartbleed...

+

=

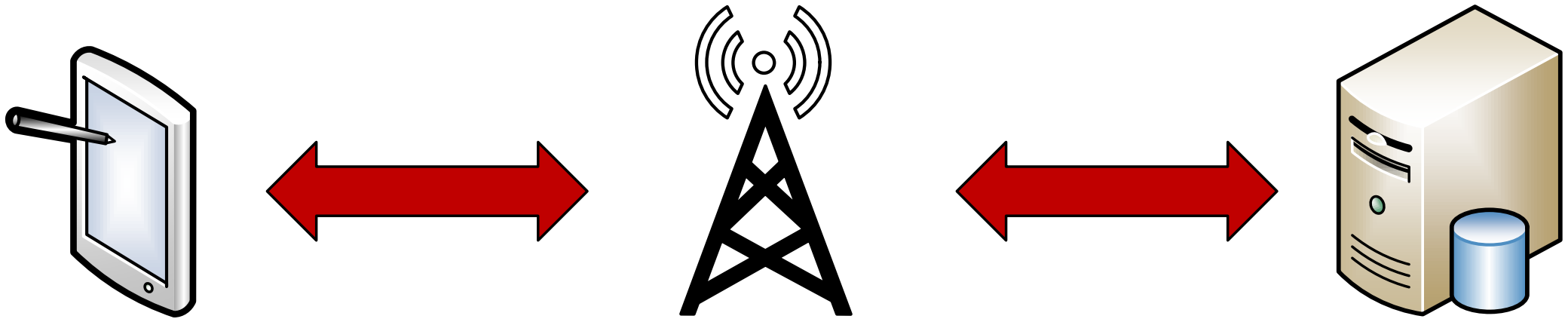
?

SAY HELLO TO “CUPID”

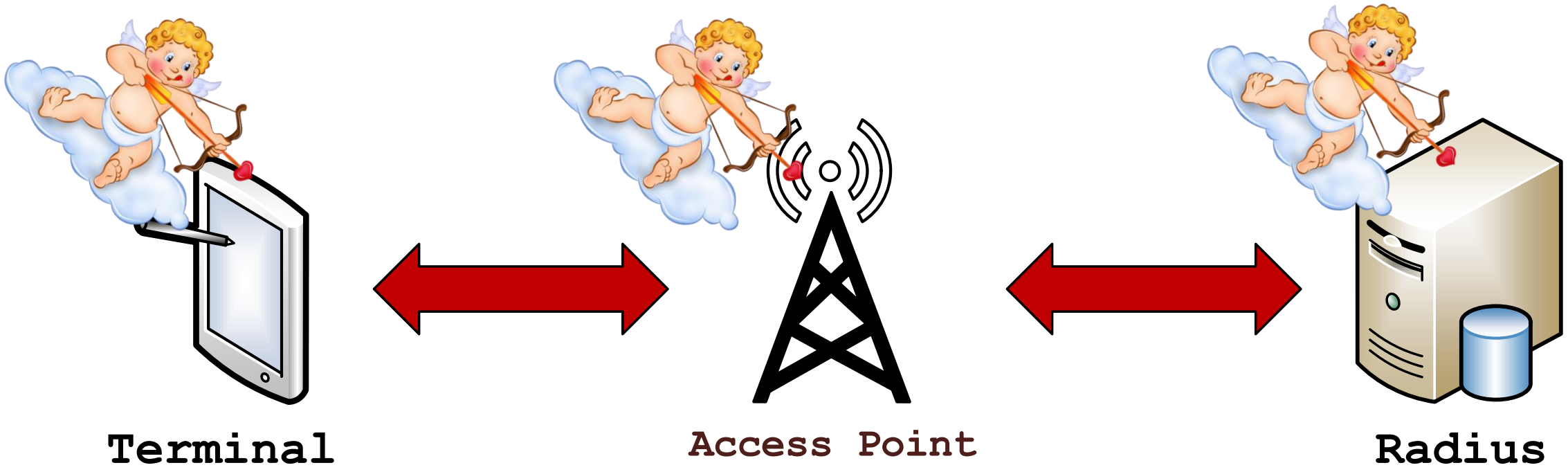
WHAT IS CUPID

- **cupid** is a patch for wpa_supplicant and hostapd
- Attempts to exploit heartbleed over EAP TLS tunneled protocols:
  - EAP-PEAP, EAP-TLS, EAP-TTLS
- Targets both endpoints: client and server

ATTACK VECTORS



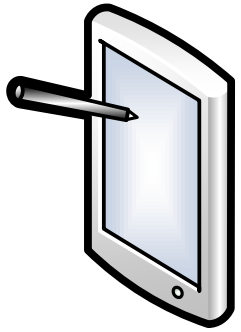
ATTACK VECTORS



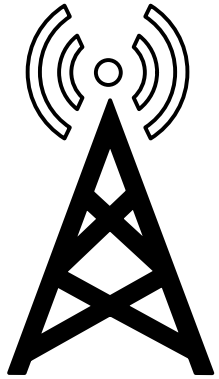
ATTACK VECTORS

- Option 1: Use **wpa_supplicant-cupid** to attack a wireless network
- Option 2: Set up a fake wireless network with **hostapd-cupid** to attack a vulnerable terminal

ATTACK VECTOR 1



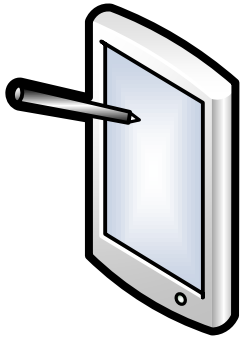
heartbleed



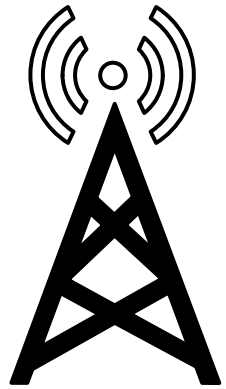
Evil client
(wpa_supplicant-cupid)

Vulnerable Access Point

ATTACK VECTOR 2



heartbleed



Vulnerable client

Evil Access Point
(hostapd-cupid)

TECHNICAL DETAILS

- Patch is able to heartbleed at different stages:
  - before TLS Handshake (unencrypted!)
  - After TLS handshake and before application data
  - After application data

DEMO TIME

VULNERABLE STUFF (CONFIRMED)

- **wpa_supplicant**
  - Android terminals, Linux devices
- **hostapd**
- **freeradius**

Must (obviously) be linked to vulnerable openssl version

VULNERABLE STUFF (POSSIBLY)

- **Everything that might use openssl for EAP TLS**
- iPhone, iPads, OSX?
- Managed Wireless Solutions:
  - Aruba, Trapeze, Cisco / Meraki...
- Other RADIUS servers besides freeradius
- Other wireless endpoints supporting EAP:
  - VoIP Phones, printers...
- Must test everything! Or patch.

ENTERPRISE “WIRELESS” && “WIRED”

- 802.1x Wired Authentication (aka NAC) uses EAP also!
- Actually, wpa_supplicant is also used on Linux to access NAC-controlled **wired** networks

CUPID AVAILABILITY

- Ask me for source code in private (for damage control)
- Will (maybe) wait a few days before releasing to public
- Tip to vendors: do not expect responsible disclosure for an exploit to a vulnerability that’s almost 2 months old...

LESSONS LEARNED

- OpenSSL sucks
- Learned a bit more about:
  - TLS Protocol
  - EAP Protocol
- Sacred cows killed:
  - “heartbleed can only be exploited over TCP connections”
  - “heartbleed can only be exploited after TLS handshake”

FUTURE WORK & RECOMMENDATIONS

- Improve patch and test ALL the things!!
  - Try different wireless devices
  - Compile and run wpa_supplicant-cupid on Android device
  - Look more closely for interesting bits of memory leaked.

- Patch Wireless clients & Servers!

THANK YOU!

lgrangeia@sysvalue.com

twitter.com/lgrangeia