Usenix 2002 Freenix Track presents

A story of a failed project

that led to a free 802.1x implementation

by Pekka Nikander

Ericsson Research Nomadiclab & Helsinki Institute for Information Technology

`http://www.tml.hut.fi/~pnr/EAPOL/`

Outline

- The story
  - Once upon a time...
  - The Great Idea
  - The Rise and Fall of iPoints
- The big picture
  - Problems in link layer security
  - 802.1x and friends
  - The fallacy of everybody being trustworthy
- Current status of the project
  - The FreeBSD 802.1x implementation
  - The structure of an integrated access point
- Conclusions

Once upon a time...

- ... or on March 28 2001, to be more exact,
- I was reading the *Theory of Money and Credit,* by Ludvig von Mises, first published in 1934, and
- I happend to start drinking beer with a few colleagues of mine...
- ... and a great idea was born ...

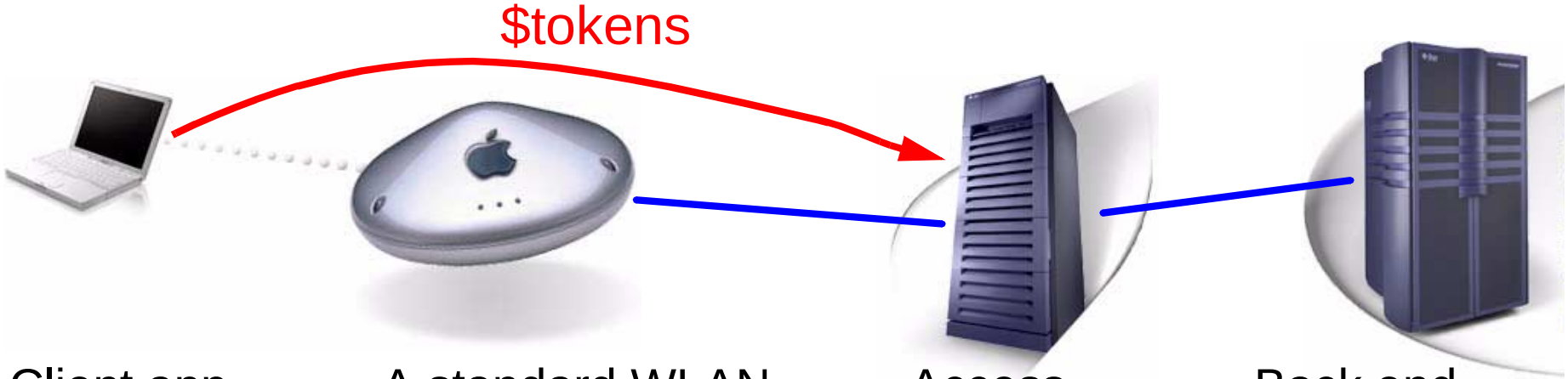
The Great Idea

(or perhaps it wasn't that great, after all, but let's not care about that)

- Public WLAN Internet access based on payment tokens
- A micro economy of tokens
  - a client sends tokes to the base station and
  - the base station owner may use them elsewhere.
- A captive portal where you can
  - download the payment software and initial tokens,
  - and buy new tokens as you need.
- Eventually run in *every* WLAN base station

- What do *we* do then? We mint the tokens!
  - The idea was to run a clearing house of the tokens

The Idea



- Laptop pays for access with tokens, sent to the firewall
- The access firewall could be
  - Integrated into the basestation (for SoHo)
  - A separate box using 802.1x to the access point
- Revenue could be *directly* split between several parties

The Rise and Fall of iPoints

- So we went and applied for seeding money
- leading to frantic business plan and prototype writing
- until one Thursday night shortly after 9/11
- our financier announced that the story was over.

- There was an unfinished business plan
- There was an unfinished prototype
- There was a bunch of engineers but no management
- There was no money

- So, that story burned out (as I almost did, too)
  - but something rose from the ashes…

The big picture

- How to do Secure Decentralized Public WLAN access?

- Issue 1: The role of trust
  - We were trying to run a bank, anyway
  - and banking is a form of monopolized trust.
  - But that’s for a completely different talk

- Issue 2: Security of public WLAN access
  - Problems in link layer security
  - How does 802.1x (and friends) fit into the figure
  - Some peole are still untrustworthy, even if they pay

Problems in link layer security

- Consider an open wireless link, e.g. public WLAN
- There are number of potential threats
  - A malicious node can masquerade as a router
  - Traffic can be directed or later redirected to nowhere or to a wrong or malicious node
  - A malicious node can feed in bad routing information
  - A malicious node can prevent other nodes from getting IP addresses, etc
- On a higher level
  - Only you pay but an attacker gets access, too
  - You are prevented from gaining access at all
  - You are lured to e.g. more expensive access

802.1x and friends

- IEEE 802.1x is an IEEE protocol for LAN authentication
  - Based on Extensible Authentication Protocol (EAP)
  - Runs anything that EAP supports, and has the same pitfalls as EAP has (and some more)
  - Works well on secure point-to-point links, e.g. switched Ethernet
- IEEE 802.11i is an IEEE protocol for WLAN security
  - Defines Robust Security Network (RSN)
  - Uses 802.1x for authentication and key distribution
  - Supports AES; with AES, provides packet integrity
- IETF PANA WG tries to define an IP layer solution
  - Still in requirements discussion phase

The fallacy of everybody being trustworthy

- Let's suppose that we have solved the link layer *authentication* and *access control* problems
  - Consequently, only authenticated hosts have access
  - Note: authentication does not imply trustworthiness!
- The local link is *still* a shared medium
  - It may be *useful* to run the link in an semi-open mode
  - Even when 802.11i AES is used, there is just one broadcast/multicast key per link
- ARP and IPv6 Neighbor Discovery (ND) depend on broadcast or multicast
- Consequently, even 802.11i does not necessarily protect against ARP/ND spoofing and other attacks
- Work on IPv6 ND starting at the IETF, BoF in Yokohama

Outline

- The story
  - Once upon a time...
  - The Great Idea
  - The Rise and Fall of iPoints
- The big picture
  - Problems in link layer security
  - 802.1x and friends
  - The fallacy of everybody being trustworthy

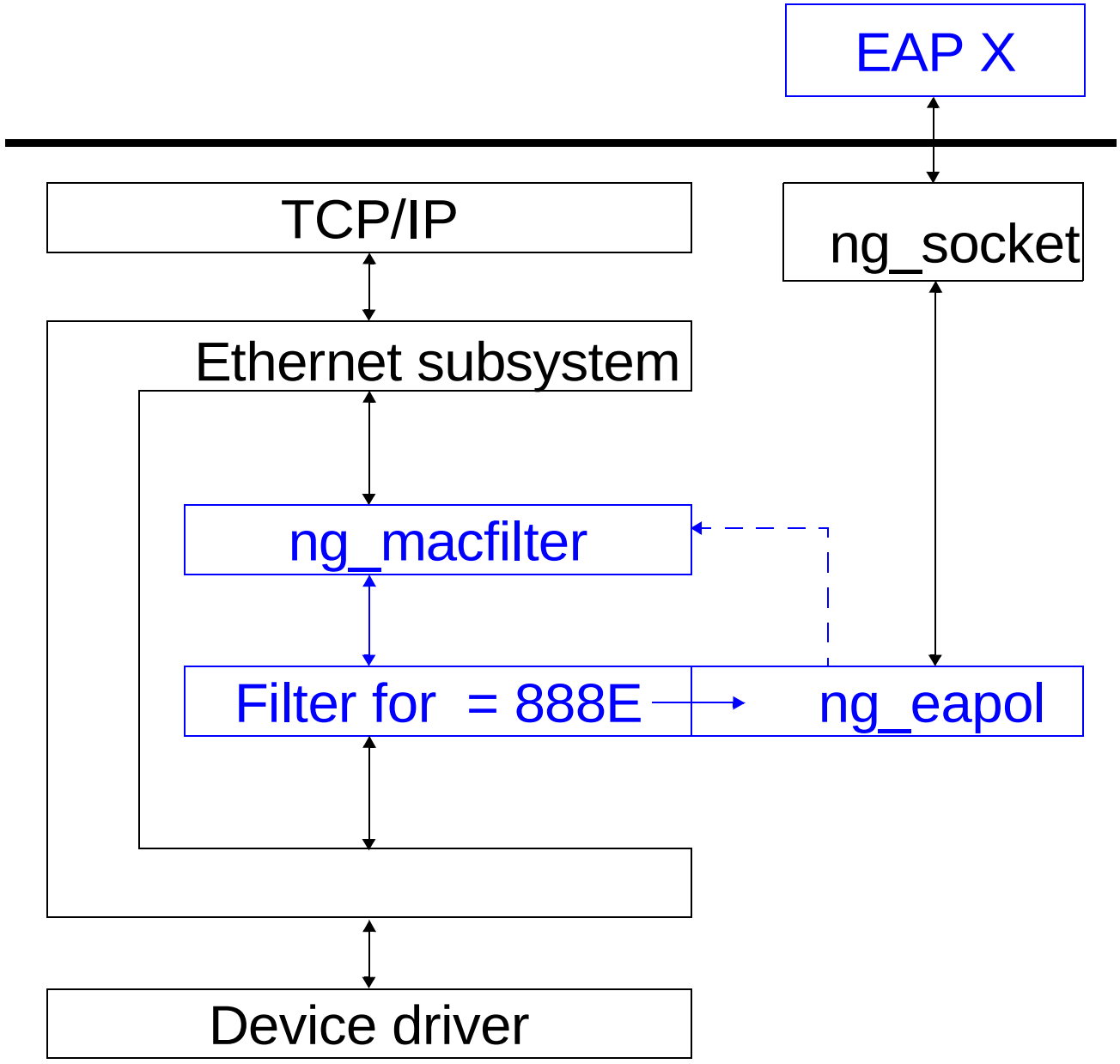
→ Current status of the project

  - The FreeBSD 802.1x implementation
  - The structure of an integrated access point
- Conclusions

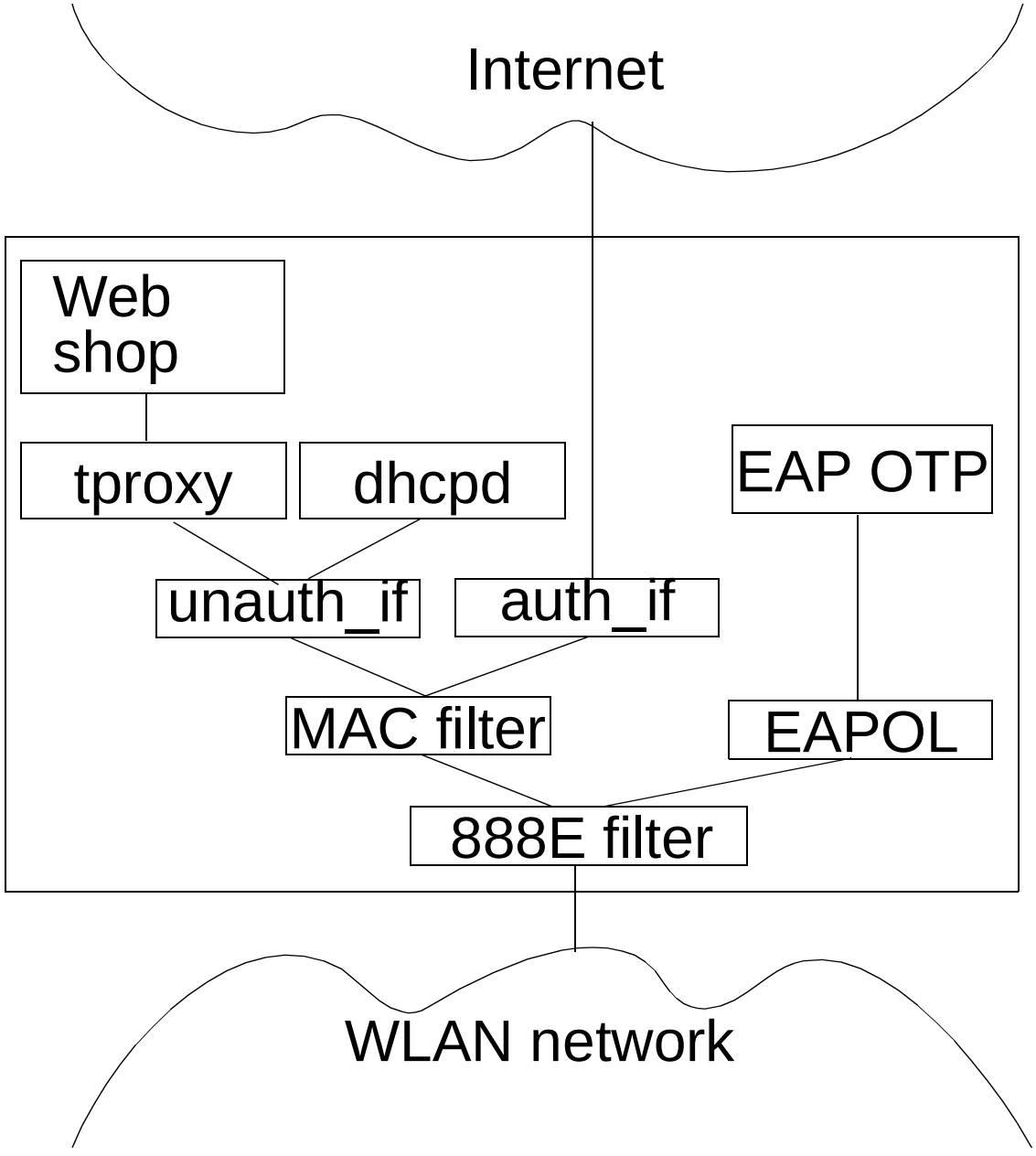
Current status

- Working FreeBSD based 802.1x implementation
  - Architecture described in next two slides
- Implemented as two netgraph modules
  - **`ng_eapol`** for EAP over LAN, the hearth of 802.1x
  - **`ng_macfilter`** for 802.1x port control
- EAP subprotocols implemented at user level
  - Currently only support OPIE
  - Writing new subprotocols is fairly straightforward
- Some features still missing
  - Mainly support for multiaccess links
- Consider this as an alpha release
  - The code is beautiful, but partially unfinished

The FreeBSD 802.1x implementation



The structure of an integrated access point



Conclusions

- Authentication and access control are a tricky problems in public wireless networks
  - Be careful with your threat and trust models
- 801.1x together with 802.11i is designed to solve the WLAN authentication and link level security
  - It does not *seem to* protect against ARP spoofing or IPv6 ND attacks
- 802.1x now implemented for FreeBSD with netgraph
  - Currently alpha level code, some features missing
  - Available right now at

    `http://www.tml.hut.fi/~pnr/EAPOL/`