



BlackHat[®]

USA • EUROPE • ASIA

USA 2004

WaveSEC for Windows

By Paul Wouters
<paul@xelerance.com>



BlackHat®

USA • EUROPE • ASIA

USA 2004



- Xelerance maintains and develops Openswan, the Linux Ipsec software.
- Continuation of the FreeS/WAN project (now defunct)
- Adopted by Debian, SuSe/IBM, Novell, Astaro.

Overview presentation

- Part one: Current '*secure*' Wireless networking
 - Deployments,
 - Protocols
 - other problems.
- Part two: Our WaveSEC solution explained
 - Building your own secure Access Point on a mini-PC
 - Putting it all in a €100 consumer AP, the Linksys WRT54g
 - Demonstrate how you can use the BlackHat WaveSEC AccessPoint.



BlackHat®

USA • EUROPE • ASIA

USA 2004

Why do we need an (Opensource) secure AP?

- April 7th 2004: <http://www.cisco.com/warp/public/707/cisco-sa-20040407-username.shtml>

*"A **default** username/password pair is present in all releases of the Wireless LAN Solution Engine (WLSE) and Hosting Solution Engine (HSE) software. A user who logs in using this username has complete control of the device. This username **cannot be disabled.**"*

Why do we need an (Opensource) secure AP?

- October 17th 2003:

<http://www.computerworld.com/securitytopics/security/story/0,10801,86187,00.html>

Joshua Wright, the systems engineer who created a tool that **targets** wireless LANs protected by Cisco Systems Inc.'s Lightweight Extensible Authentication Protocol (**LEAP**), said he did so to demonstrate the ease with which dictionary attacks against the protocol can **crack user passwords**.

Wright said Cisco users should "be aware of the risks that exist by using the LEAP protocol." He said he plans to release the attack tool, which he has dubbed ASLEAP, in February, although he declined to say how he would make it available.

The tool uses a challenge-and-response methodology built into LEAP to obtain the information needed to mount a dictionary attack, according to Wright. He then uses a 100GB electronic dictionary that includes various languages to **discover passwords**, a process that Wright said can be done **in a matter of seconds**.

- Cisco released advisory on april 12th 2004 (5 months later!)

<http://www.cisco.com/warp/public/707/cisco-sn-20030802-leap.shtml>



BlackHat®

USA • EUROPE • ASIA

USA 2004

Why do we need an

- May 13th 2004: <http://www.auscert.org.au/render.html?it=4091>
(Opensource) secure AP?

Denial of Service Vulnerability in IEEE 802.11 Wireless Devices

“ An attacker using a low-powered, portable device such as an electronic PDA and a commonly available wireless networking card may cause **significant disruption** to all WLAN traffic within range, in a manner that makes identification and localisation of the attacker difficult.”

“ At this time a comprehensive solution, in the form of software or firmware upgrade, is not available for retrofit to existing devices. Fundamentally, the issue is **inherent in the protocol** implementation of IEEE 802.11 DSSS.”

Why do we need an (Opensource) secure AP?

- May 4th 2004: <http://www.uniras.gov.uk/vuls/2004/236929/>

Vulnerability Issues in TCP

The issue described in this advisory is the practicability of resetting an established TCP connection by sending suitable TCP packets with the RST (Reset) or SYN (Synchronise) flags set.

"The Border Gateway Protocol (**BGP**) is judged to be potentially most affected by this vulnerability."

Why do we need an (Opensource) secure AP?

- April 20th 2004: <http://www.uniras.gov.uk/vuls/2004/236929/>

The following mitigation steps are still being evaluated and may be incomplete. Customers should work with vendors for the workaround most appropriate for the product in question [...]

- Implement IP Security (**IPSEC**) which will encrypt traffic at the network layer, so TCP information will not be visible.
- Reduce the TCP window size (although this could increase traffic loss and subsequent retransmission).
- Do not publish TCP source port information.

New problems

- Various new wireless communication protocols (Bluetooth, GPRS, GSM, WDCMA, WiFi)
- New billing models for hotspot access (scratch cards, subscriptions, roaming)
- Wireless is much easier to eavesdrop than ethernet cables or phonelines
- Connecting to a rogue Access Point; Or accidentally connecting to a private Access Point
- You have to be able to connect to the network before you can authenticate, pay and then somehow go into a secure mode to use the Access Point.
- Most standard way of securing Access Points is WEP, which is useless for hotspots, since you are telling everyone all the secrets (The WEP key)
- You can't rely on preloaded software by a sysadmin, since this might be a roaming user.

New Markets: Lots of money to be made NOW

- Bind users through AccessPoint capabilities
- Bind users through Wireless card capabilities
- Bind users through Certification Systems
- Grabbing new customers is more important than security
- Binary only firmware to protect Intellectual Property
- Binary only firmware to restrict radio access (FCA requirement)



BlackHat®

USA • EUROPE • ASIA

USA 2004

Security vs Marketing

New solutions often based on hype:

- Focus on desirable billing method (Get rich quick)
- Focus on customer 'relationship' (Get rich quick)
- Focus on pushing users through portals (Advertisement income), sometimes preventing users from full access.
- Cheap uplink, almost always behind NAT
- Often heard excuse: New protocols need to work on old AP hardware.
- Strange desire to “protect the link layer”



BlackHat®

USA • EUROPE • ASIA

USA 2004

Security vs Marketing

Classic solutions often based on perfect security

- Not lightweight solutions (problem for PDAs and APs)
- Require complex software and cryptography
- Require extensive CS knowledge to configure for use
- Require pre-arrangement or trusted third party to prevent “man in the middle” attacks, which goes against commercial desire to quickly take customers
- Too much is in Microsoft's hands (no Windows, no go)

WiFi Standards slowly emerging

- WEP: old 128bit, weak IV broke most WEP implementations:
<http://wepcrack.sourceforge.net/>
- WEP+: fixed weak Ivs, 256bit, but it is still WEP
- WPA: worse then WEP for passphrase of less then 20 characters: <http://wifinetnews.com/archives/002453.html>
Supported by Microsoft, more difficult with other OS.
- EAP: Extended Authentication Protocol. Many new layers to protect, layers “carry over” from previous crypto processing. Complex. Not unlikely to get broken. Projects to connect EAP with SIM and Radius, see <http://www.wlansmartcard.org/>

WiFi Standards slowly emerging

- LEAP: cracked 9 months ago, withheld by Cisco
<http://asleep.sourceforge.net/>
- PEAP: Son of LEAP, less patents than LEAP, more secure.
For now...
- 802.1x (don't confuse with 802.11x): EAP-Radius based. See
<http://www.open1x.org/>
- Dynamic WEP: often combined with 802.1x

Problems: Most of them operate in the card, so binary firmware only. Makes it more difficult to fix or upgrade too.

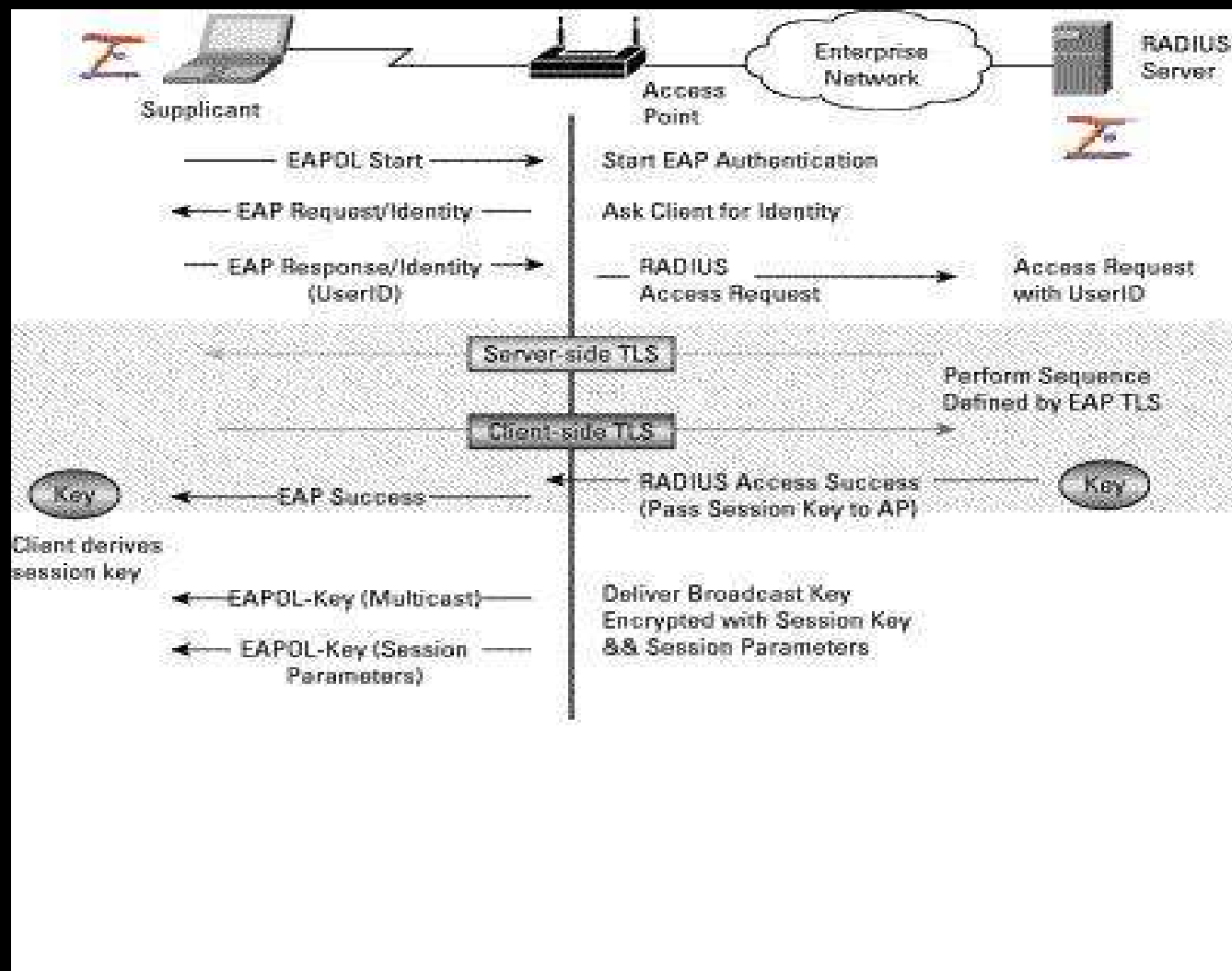


Black Hat®

USA • EUROPE • ASIA

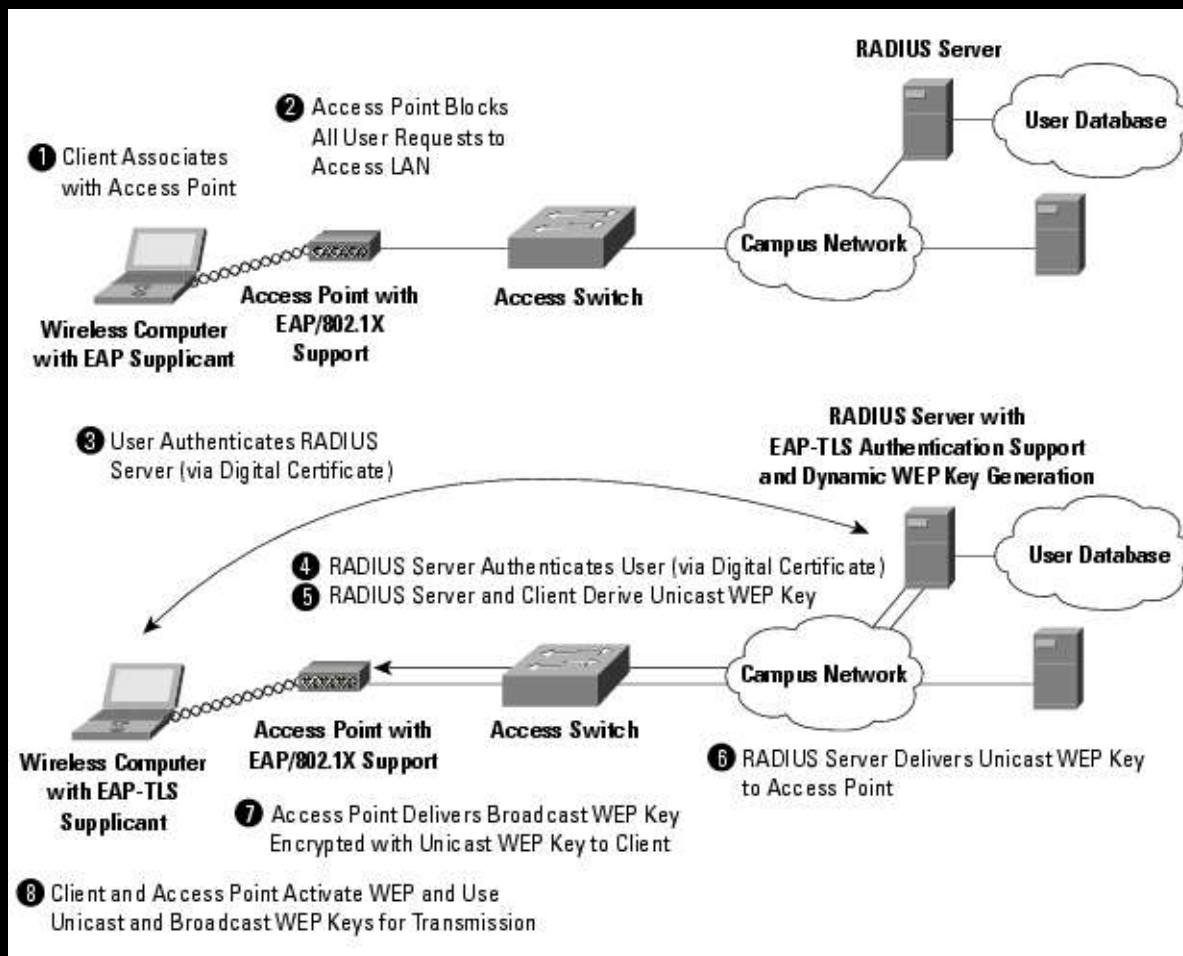
USA 2004

Complexity of EAP





Complexity of EAP



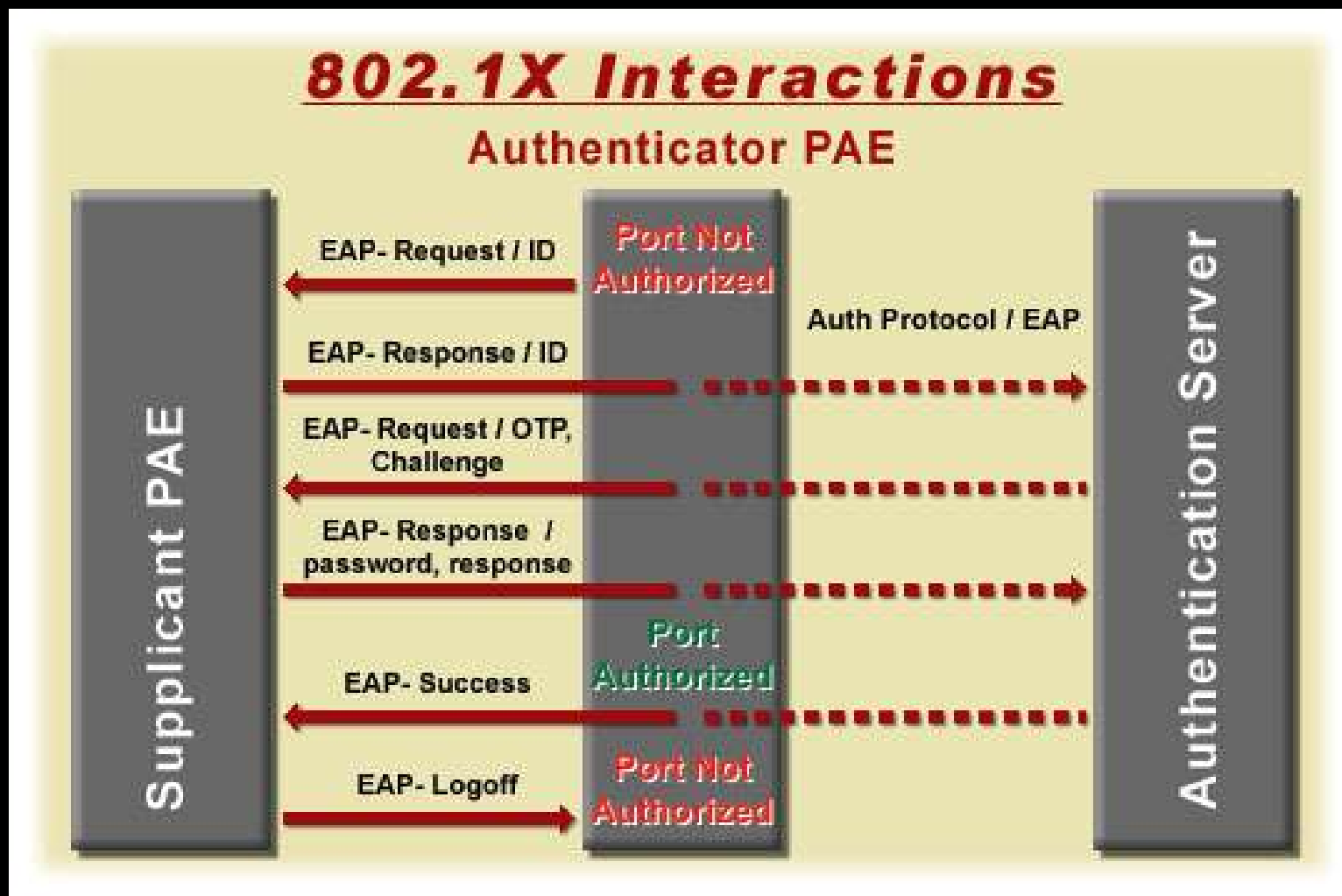


Black Hat®

USA • EUROPE • ASIA

USA 2004

Complexity of EAP





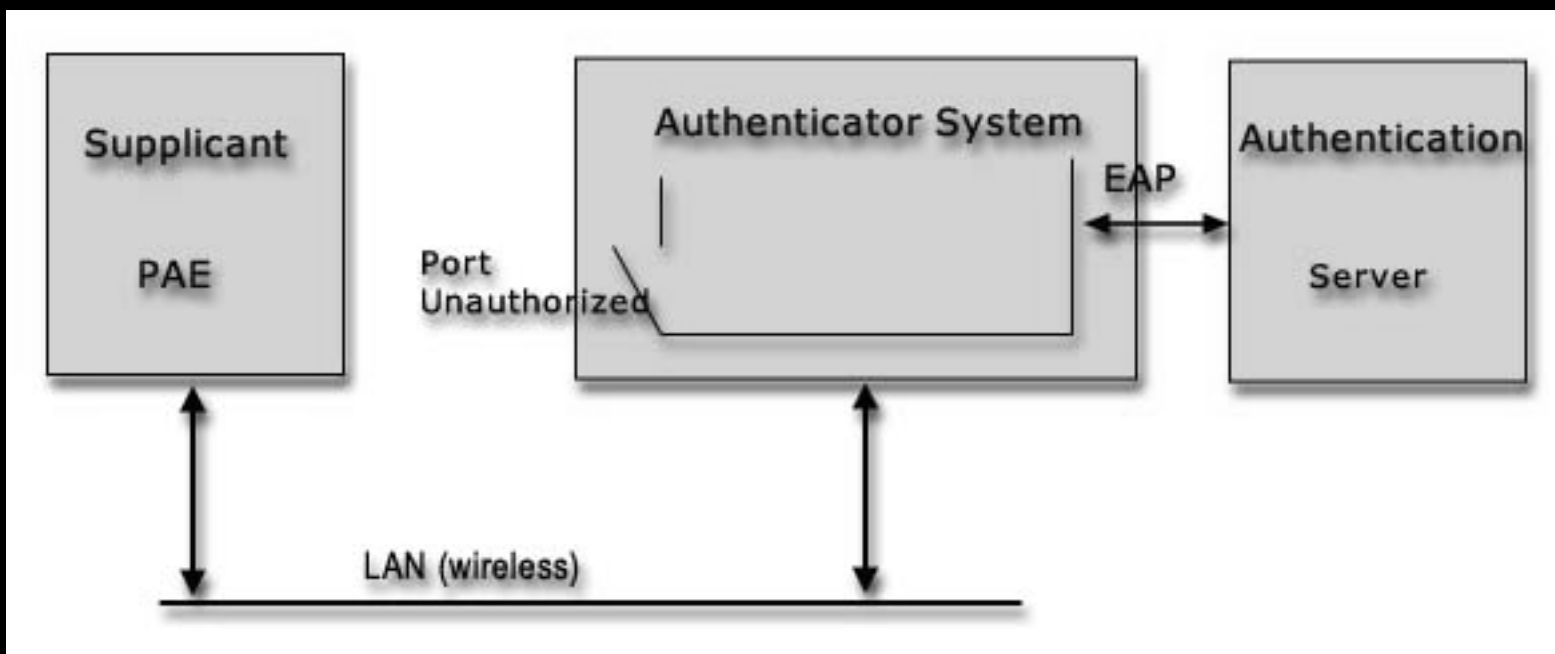
BlackHat®

USA • EUROPE • ASIA

USA 2004

802.1x

- Windows has driver support.
- Linux support is poor; Missing Cisco and Centrino
- Hacks using Win32 binary: DriverLoader and ndiswrapper.



VPN standards emerging

- SSL based VPNs: Low Latency, Vulnerable to RST attacks)
- Custom VPN clients: Nortel, Cisco, Windows (hardly interop, Usual broken behind NAT)
- Unix hacks: stunnel (see above), CIPE (cracked)
- Microsoft hack: LT2P (IPsec with glue to use RAS)
- IPsec with RFC extensions
 - X.509: Certificates
 - XAUTH: user/password
 - IKEv2 (Advanced options negotiations)

What is a “ hotspot”

- Redirect all traffic to authentication site (usually AP)
- Authenticate user, do billing
- (optionally?) encrypt all traffic
- Stop redirecting user (redir over proxy instead)
- De-authenticate when EO\$

- Redirection to authentication server is vulnerable to MITM
- AP can be spoofed by malicious user

What not to protect

- We cannot protect against users associating with a rogue Access Point as long as we do not have cryptographically secured beacons.
- We cannot protect the link layer.
- Protect against DoS as much as we can (limit use of TCP 3way handshake, try to use Ipsec)
- EAP/802.11 alone cannot fix this. IPsec with authentication can. It could even use EAP/802.11, but why? There are other ways.



BlackHat®

USA • EUROPE • ASIA

USA 2004

Our proposal: WaveSEC

- Use proven technology: IPsec with either X.509 or DNSSEC/DHCP
- Don't care about the link layer. Enforce crypto, do authentication in IP layer (“ There is no OSI model”)
- IPsec supported by most network devices
- IPsec has been deployed widely, and has not been broken in many years.
- No patents, licences, royalties or binary-only software or firmware
- Possibility to separate WiFi and Crypto operations, so that the radio, or even AP, doesn't need to do the crypto operations that are CPU expensive

IPsec in a nutshell

- Part 1: Diffie-Hellman Key Exchange
 - Ensures privacy
 - Vulnerable to Man in the middle attack
- Part 2: Identity exchange and verification
 - Exchange ID's
 - Both parties independantly check ID with trusted third party (dnssec or CA).
 - Both parties agree on encryption method, eg RSA key based. RSA key of other party needs to be signed with a known and trusted CA.
 - Both parties agree on a stream cipher for the encryption, eg AES
 - Both parties agree to pass along certain packets, eg 10.0.1.0/24
 - Extra's: NAT Traversal, Dead Peer Detection, XAUTH/RADIUS,

Unresolved problem by all technologies

- Rogue Aps. Users cannot control which AP they associate with. Rogue AP means rogue DHCP and/or rogue SSL.
- Trusted third party. Users have to make some leap of faith at some point, unless they pre-arrange something (DNSSEC is not deployed yet, CAs are too trivial to inject or falsify)
- With IPsec, at least if you do switch later on, you only send the rogue AP crypted garbadge.

Misconceptions about WaveSEC

- TALKING SECURELY TO A NEW HOST REQUIRES A 3RD PARTY PROVIDING CREDENTIALS !!!

This can be:

- Recognised and trusted Certificate Agency (trusted root CA)
- DNSSEC resolution from a Secure Entry Point (SEP)
- An enduser manually verifying the cryptographic key using a fingerprint.
- Ssh-style 'Leap of Faith' (caching new keys to verify) (also known as 'Me Tarzan, You Jane')



BlackHat®

USA • EUROPE • ASIA

USA 2004

Wireless connectivity options

- Do not use cryptography at all
- Vulnerable to all passive attack
- Vulnerable to local network active attacks (rogue AP, rogue DHCP, rogue DNS, etc)
- Vulnerable to remote network active attacks (Man in the middle attack to remote servers from LAN)

Not recommended!!!

Wireless connectivity options

- Use the provided proprietary vendor specific WiFi protocol security (LEAP, WPA, WEP, etc)
- Most crypto either broken (WEP, WPA, LEAP) or haven't had a long peer review in the crypto community yet.
- Protects against passive attacks
- Vulnerable to local active attacks (eg rogue AP supporting WPA)
- Vulnerable to remote attacks

Wireless connectivity options

- Use Wavsec (Opportunistic Encryption) with DNS using IPsec
- Does not use weak or broken or untested proprietary crypto protocols but rigourously tested IPsec protocols.
- protects against passive attacks
- Initially vulnerable to active attacks using rogue Access Points, or DHCP/DNS servers, but only towards other local LAN wavsec clients if enduser does not verify manually.
- Not available for Windows or MacOSX
(port of Openswan to MacOSX is planned)

Wireless connectivity options

- Use Wavesec (X.509) certificates with IPsec
- Does not use weak or broken or untested proprietary crypto protocols but rigourously tested IPsec protocols.
- Protects against passive attacks
- Protects against active attacks using rogue Access Points, or DHCP/DNS servers.
- Needs trusted third party CA verification and manual verification (tedious and user unfriendly, most users will just click OK anyway)



BlackHat®

USA • EUROPE • ASIA

USA 2004

Wireless connectivity options

- Use Wavesec (OE) with Ipsec and DNSSEC
- Does not use weak or broken or untested proprietary crypto protocols but rigourously tested IPsec protocols.
- Protects against passive attacks
- Protects against all active attacks
- Needs some manual setup for SEP's until DNSSEC becomes widely deployed, but when deployed on a large scale is a fully automated secure process without any user interaction (no stupid users clicking OK anyway)
- Not yet available for Windows or MacOSX

Imminent developments

- IETF: DNSEXT working group is finalising DNSSEC-bis internet-drafts so they can go to IESG to become RFC's.
- IETF DHC working group plans to use DNSSEC to protect DHCP protocol against rogue DHCP servers
- IETF: IKEv2 The new version of IKE, the Internet Keying Exchange protocol for IPsec will include Opportunistic Encryption type hooks. This will move part of our current DHCP additions within the IKE protocol, which is then both hidden and protected by the ISAKMP Security Association.



BlackHat®

USA • EUROPE • ASIA

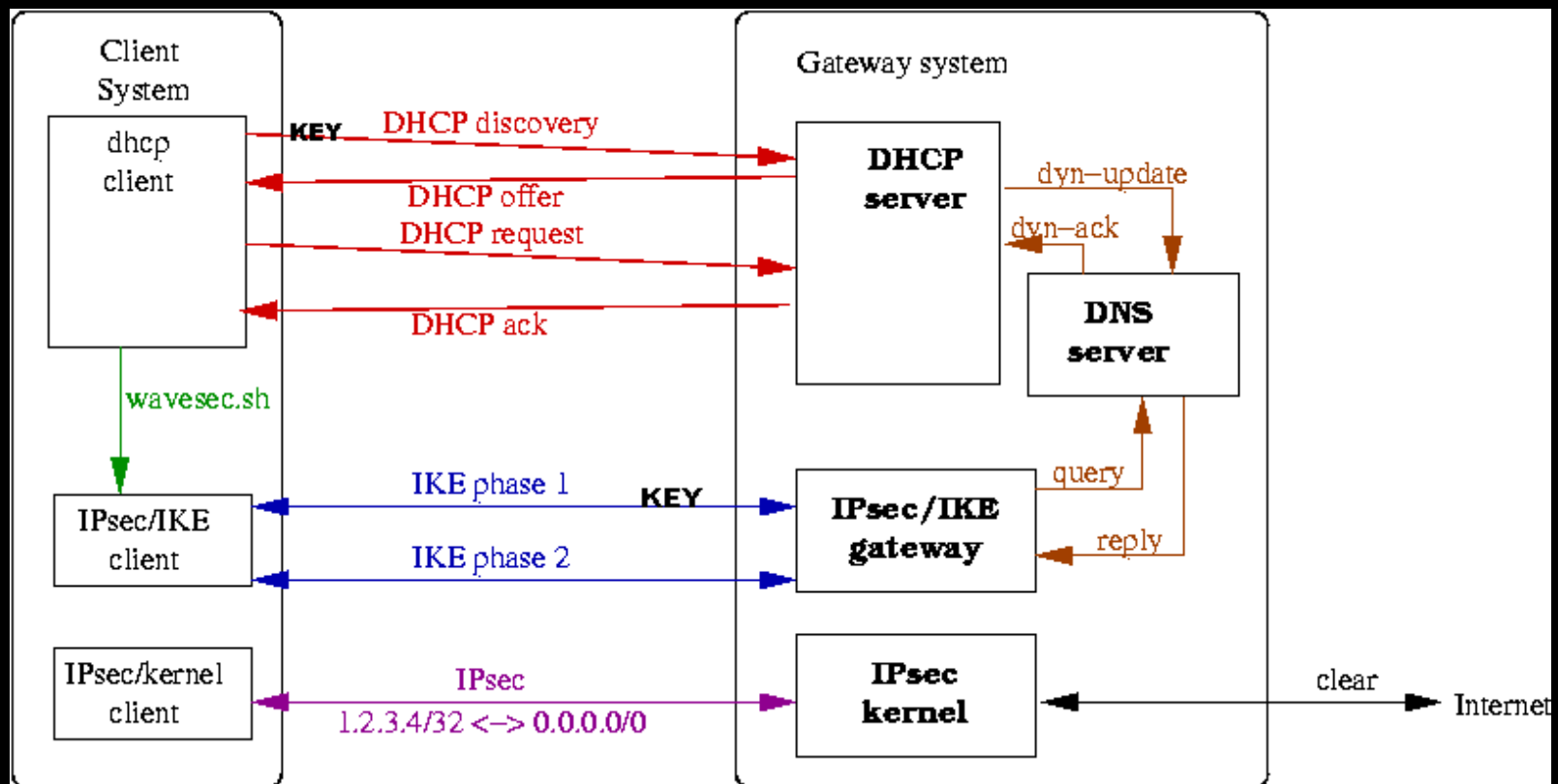
USA 2004

Coffee Break





WaveSEC for full IPsec clients (UNIX)



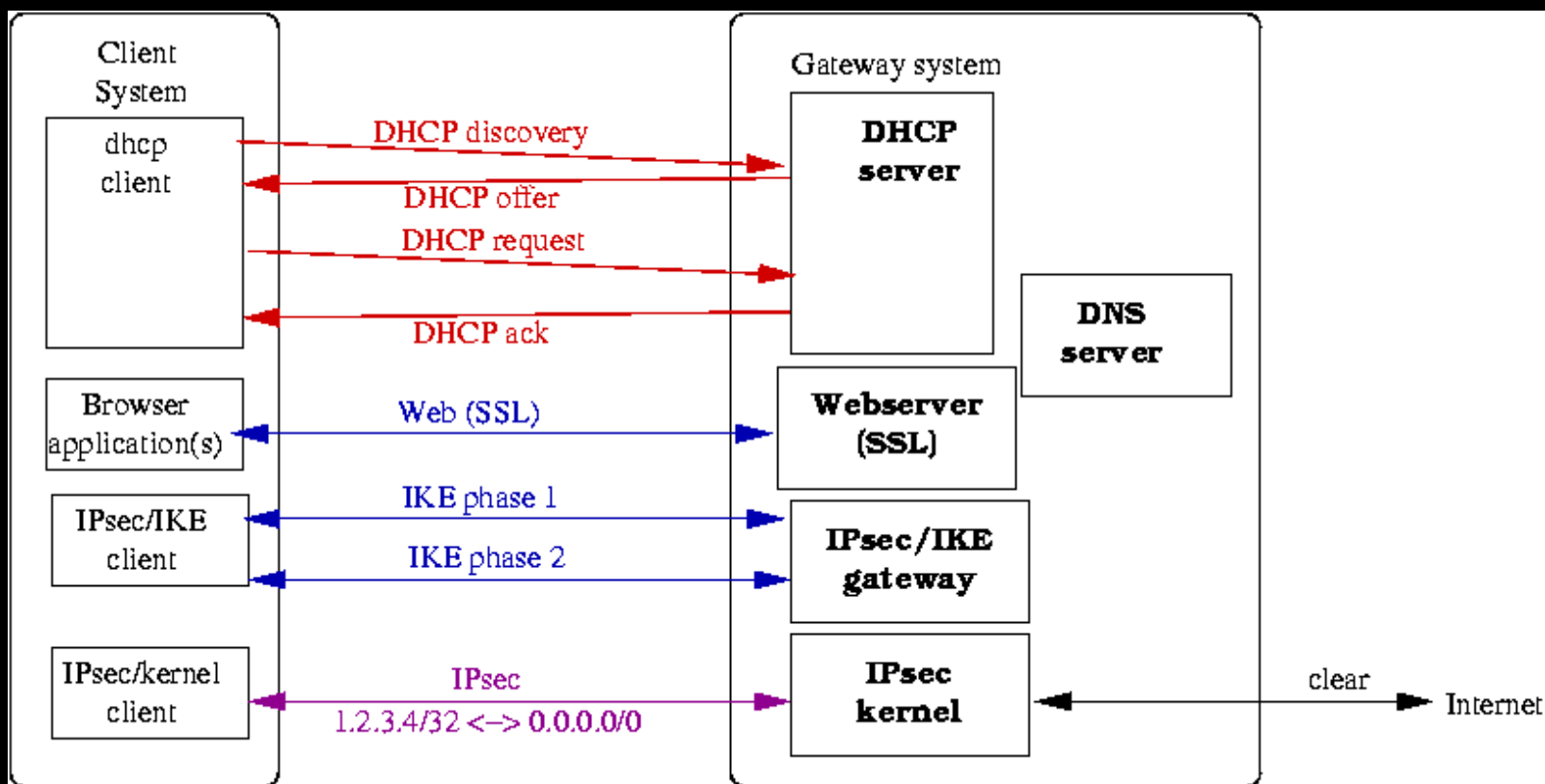


Black Hat®

USA • EUROPE • ASIA

USA 2004

WaveSEC for Windows clients





BlackHat®

USA • EUROPE • ASIA

USA 2004

Building your own Access Point with WaveSEC

- Provide a DHCP server (ISC dhcpd)
- Provide a DNS server (ISC bind9)
Good idea to ratelimit dns packets to prevent people using IP-over-DNS tunneling, eg <http://nstx.dereference.de/>
(don't tell StarBucks or Krasnapolsky)
- Provide an IPsec server (Openswan)
 - X.509 certificate generation on the fly after CreditCard processing?
 - XAUTH/Radius based scratch cards?



BlackHat®

USA • EUROPE • ASIA

USA 2004

Building your own Access Point with WaveSEC

- Provide SSL capable webserver (Apache)
 - For downloading custom software, and explain the user what to do.
- Provide X.509 functionality (OpenSSL)
 - for generating CA, certs and signatures.
- Provide Transparent Proxy server (Squid w. IPtables)
 - makes AP seem faster



BlackHat®

USA • EUROPE • ASIA

USA 2004

WaveSEC prototype

- Symtrax Cyrix MediaGX 300mhz, 64MB RAM, 20GB disk, 3x ether.





WaveSEC prototype software based on Fedora

- Full RedHat Fedora Core 1 install
- Used RPMS for apache,openssl, dhcpd,php
- Used Openswan-2 (ftp.openswan.org)

We glued everything together using PHP and Expect

WaveSEC prototype: Generate CA

- “Initialise Certificate Agency” button
- `mkdir /etc/sslca ; cd /etc/sslca`
- edit `/usr/share/ssl/openssl.cnf` to taste (eg name, default_bits, change default path from demoCA to `/etc/sslca`, change validity (3650 days))
- `/usr/bin/openssl req -x509 -days 1460 -newkey rsa:1024 -keyout caKey.pem.locked -out caCert.pem -passin pass:foobar -passout pass:foobar`

WaveSEC prototype: Generate AP key

- `/usr/bin/openssl req -newkey rsa:1024 -keyout filename.Key.pem.locked -out filename.Req.pem -passin pass:foobar -passout pass:foobar`
- Optionally remove passphrase for software
- `openssl rsa -passin pass:foobar -passout pass:foobar -in filename_lock -out filename_unlock`

WaveSEC prototype: Sign & Install AP key

- `/usr/bin/openssl ca -in filename.Req.pem -days 730 -out filename.Cert.pem -passin pass:foobar -notext -cert caCert.pem -keyfile caKey.pem.locked`
- `cp gatewayCert.pem /etc/ipsec.d/certs/ # AP host pubkey`
- `cp gatewayKey.pem* /etc/ipsec.d/private/ # AP host privkey`
- `cp caCert.pem /etc/ipsec.d/cacerts/ # AP host cert CA`
- `# following needs entry in /etc/ipsec.secrets`
`cp gatewayKey.pem.locked /etc/ipsec.d/private/`
- `# Certificate Revocation List (optional)`
`openssl ca -gencrl -out /etc/ipsec.d/crls/crl.pem`
- `Service httpd restart ; service ipsec restart`



BlackHat®

USA • EUROPE • ASIA

USA 2004

WaveSEC prototype: Configure Openswan

- Configure /etc/ipsec.secrets
: RSA blackhat.xelerance.com.key "your_password"
- Configure /etc/ipsec.conf wavesec connection
conn wavesec-for-windows
 right=%any
 left=%defaultroute
 leftsubnet=0.0.0.0/0
 leftcert=blackhat.xelerance.com.pem
 leftid="C=NL,L=Amsterdam,O=Xelerance,OU=Wireless
 Security Department,CN=CA wireless,
 E=postmaster@xelerance.com"
 auto=add



BlackHat®

USA • EUROPE • ASIA

USA 2004

WaveSEC prototype: Configure Openswan

- Leftid option can be seen with:
openssl x509 -in cacert.pem -noout -subject
- Check and see if connection loaded correctly with:
ipsec auto --listall
(double check that “ has private key” appears with gateway key)



BlackHat®

USA • EUROPE • ASIA

USA 2004

WaveSEC prototype: Configure PHP

- Optional: Install “nocat” for port redirection to AP
- Interpret browser OS and redirect to client page:

```
include("wavesec.inc");
check_and_go_secure();           $browser =
$GLOBALS["HTTP_USER_AGENT"];
if (stristr($browser,"Linux")!= FALSE)
    Header("Location: /linux/");
else if (stristr($browser,"Windows NT 5.1")!=FALSE)
    Header("Location: /winxp/");
else if (stristr($browser,"Windows NT 5.0")!=FALSE)
    Header("Location: /win2k/");
else if (stristr($browser,"Mac OS X")!=FALSE)
    Header("Location: /macosx/");
else Header("Location:/other/");
```

WaveSEC prototype: Configure PHP

- Generate a new hostkey for the client on the AP (Identical to generating the gateway key earlier)
- Optionally remove passphrase:
`openssl rsa -passin pass:foobar -passout pass:foobar -in filename_lock -out filename_unlock`
- For windows client, an extra step, make PKCS12 file (includes root CA)
`/usr/bin/openssl pkcs12 -export -inkey filename_lock -in filenameCert.pem -name wavesec -certfile caCert.pem -caname \"WaveSEC CA\" -out filenameCert.p12 -passin pass:foobar -passout pass:foobar`

WaveSEC prototype: Making wavesec.exe

- "Our" client is made with NullSoft Installer Software (NSIS), consists of:
 - IPsec supportive tools for either XP or 2K
 - WinXP: ipseccmd.exe from WinXP CD:\SUPPORT\TOOLS
 - Win2k: ipsecpol.exe
<http://agent.microsoft.com/windows2000/techinfo/reskit/tool>
 - Ebootis VPN tool <http://vpn.ebootis.de/package.zip>
(ipsec.exe)
 - certificate loader: certimport.exe (certimport -f foobar clientXXCert.p12) <http://www.xelerance.com/>

WaveSEC prototype: Making wavesec.exe

- ipsecmon.exe for debugging (Win2k only)
- wget.exe with ssl to fetch p12 file. (For possible future use) (<ftp://ftp.sunsite.dk/projects/wget/windows/wget-1.9.1b-complete.zip>)
- ipseccmd and the MMC ipsec snap-ins for debugging (ipseccmd \\yourmachinename show all)
- We packages these files into our wavesec client files:

WaveSEC-0.99bh-xp.exe (BlackHat CD)

WaveSEC-0.99bh-2k.exe (BlackHat CD)

WaveSEC prototype: Limited experience so far

- currently, our exe files are static. We have to separately download, or let the user download the configuration file and the certificate file.
(We are working on hacking self-extracting zip files on linux)
- Prevent leaching certificate files by Evil Users. Eg: delete upon download.
(not yet implemented in prototype)
- Extend NSIS package to 'figure out' where the certificate file and Windows' ipsec.conf file were downloaded (fetch with wget? dynamically overwrite self extracting .exe files?)



BlackHat®

USA • EUROPE • ASIA

USA 2004

WaveSEC prototype: Limited experience so far

- Windows does send Notify/Delete, but Openswan ignores them. Bug?
- If Openswan ignores them (or windows box crashes and wont send them), we can have two identical conns open on different IP's. Use uniqueids=no should mitigate this (kills older client connection)
- Use rekey=no (server kills idle clients, clients have to rekey actively)
- I am also not sure "ipsec -off" properly works on Windows. Intermittent issues.



BlackHat®

USA • EUROPE • ASIA

USA 2004

WaveSEC prototype: Limited experience so far

- Windows seems to accept plain text communication for policies that should only do crypto. Windows bug or ipsec.exe policy agent bug. Need to be traced down.
- People removing WaveSEC software while policies are loaded. Yes they are loaded again after reboot, without the need for the supporting tools!!
- Windows can only tunnel “everything” to the default gateway. If fails to send packets for “everything” to another host. Though that is a fairly bad setup anyway, requiring NAT. (think “limited hotel IPs”)

WaveSEC prototype: TODO

- ipsec -off at shutdown/suspend
- get rid of dos box (make real win32 binary)
- tray icon for on/off
- splashscreen :)
- better certificate installer with file selector menu.
- Or modify self-extracting zip file so we can add certificate and configuration file at a known place within the .exe file, so know exactly where to find them to process them (eg to insert the certificat into the Registry)



Try out WaveSEC at the conference!

- Grab me during the conference if you need help
BlackHat CA cert for WaveSEC is on the BlackHat CD

```
openssl x509 -in BlackHatcaCert.pem -noout -fingerprint
```

```
MD5 Fingerprint=
```

```
02:C2:0E:04:DC:4E:92:50:EA:1B:A5:EA:D9:B0:7D:CE
```

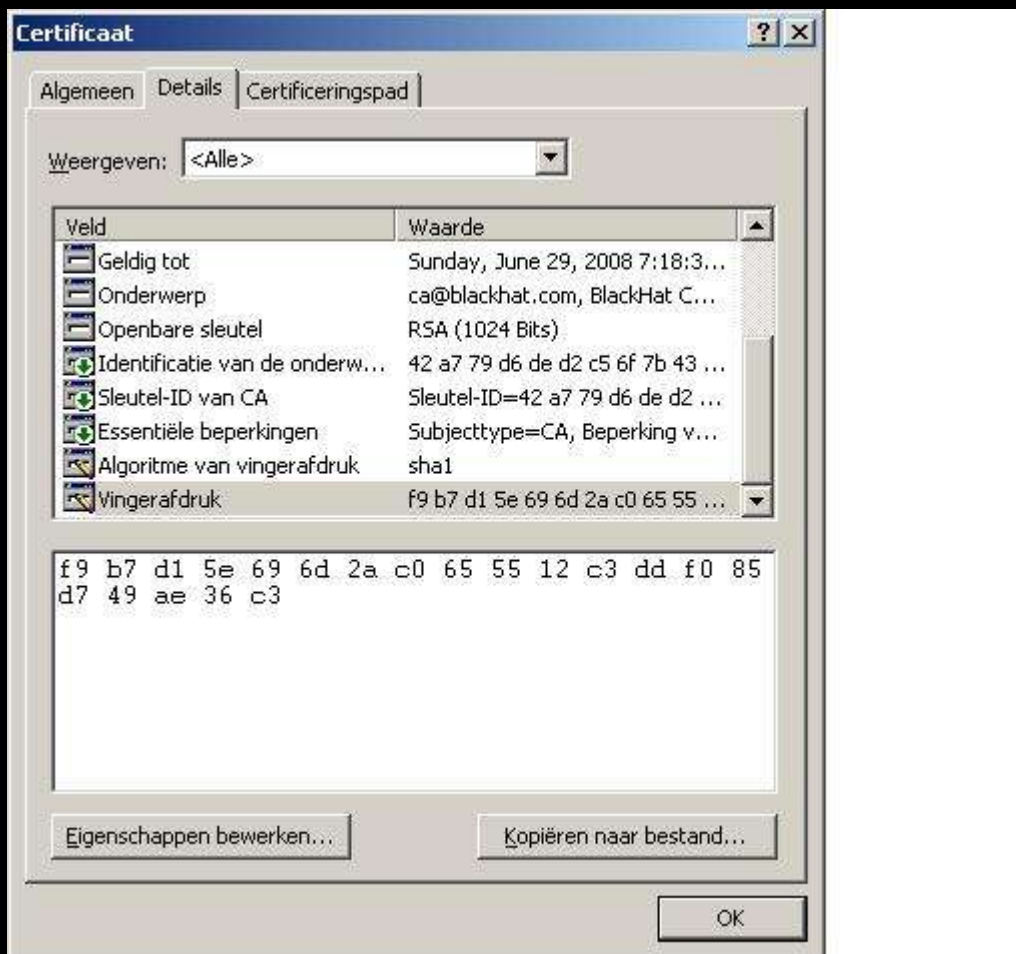


BlackHat®

USA • EUROPE • ASIA

USA 2004

Try it out at the conference





BlackHat®

USA • EUROPE • ASIA

USA 2004

Next step: WaveSEC on consumer AP

- Linksys WRT54g (100Mhz MIPS, 16MB RAM, 4MB FLASH)



Next step: WaveSEC on Linksys

- It runs Linux, and we can redo the kernel and rest of the system.
- Runs Openswan-2 (as of 2.1.2) including AES and 3DES (1000 Kbyte/sec AES encryption/decryption)
- based on OpenWRT (<http://openwrt.ksilebo.net/>)
- haven't squished it all in 16MB yet, so using nfs mount for storage
- Use "starter" instead of all the sed/awk/perl scripst to start IPsec
- Perhaps pre-calculate certificates, since the MIPS CPU isn't that good? (120Mhz MIPS on version 1 and 200Mhz on "Speedbooster")
- Look for mini SSL capable webserver (BOA? Perl? microasp?)



BlackHat®

USA • EUROPE • ASIA

USA 2004

Next step: WaveSEC on Linksys

- We ported Openswan-2 to the MIPS/openwrt platform. Patches are included in Openswan-2.1.2 (released may 19 2004)
To install, add the following to /etc/ipkg.conf:

```
src openswan ftp://ftp.openswan.org/openswan/binaries/openwrt/buildroot-20040509/ipkg/
```

and run:

- `ipkg update`
- `ipkg install gmp mawk openswan-module openswan`
- Speed: 1000 Kbyte/sec AES encryption and decryption.
- Userland has been confirmed to work with RSAkey and X.509, AES and 3DES