Wireless LAN Security with 802.1x, EAP-TLS, and PEAP

Steve Riley Senior Consultant MCS Trustworthy Computing Services



So what's the problem?

- WEP is a euphemism
 - Wired
 - Equivalent
 - Privacy
- Actually, it's a lie
 - It isn't equivalent to "wired privacy" at all!
 - How can you secure the air?
- So: WEP sucks
 - http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html

Wired equivalent privacy



WEP setup and RC4

- Secret key shared between access point and all clients
 - Encrypts traffic before transmission
 - Performs integrity check after transmission
- WEP uses RC4, a stream cipher
 - [key] XOR [plaintext] \rightarrow [ciphertext]
 - Maybe double-XOR for "better" security? Hah!
 - [ciphertext] XOR [key] → [plaintext]

Common attacks

- Bit-flipping (encryption \neq integrity)
 - Flipping bit n in cipertext flips same bit in plaintext
- Statistical attacks
 - Multiple ciphertexts using same key permit determination of plaintext XOR
 - Enables statistical attacks to recover plaintext
 - More ciphertexts eases this
 - Once one plaintext is known, recovering others is trivial

WEP's "defenses"

- Integrity check (IC) field
 - CRC-32 checksum, part of encrypted payload
 - Not keyed
 - Subject to bit-flipping → can modify IC to make altered message appear valid
- Initialization vector (IV) added to key
 - Alters key somewhat for each packet
 - 24-bit field; contained in plaintext portion
 - Alas, this small keyspace guarantees reuse

More IV problems

- Say an AP constantly sends 1500-byte packets at 11mbps
 - Keyspace exhausted in 5 hours
 - Could be quicker if packets are smaller
- Key reuse causes even more collisions
 - Some cards reset IV to 0 after initialization
 - Some cards increment by 1 after each packet
- 802.11 standard does not mandate new per-packet IV!

Classes of attacks

- Key and IV reuse
 - Small IV space; no IV replay protection
- Known plaintext attack
 - Can recover stream of length N for a given IV
 - Then forge packets of length N in absence of keyed IC
- Partial known plaintext attack
 - Can recover *M* bytes of keystream, *M* < *N*
 - Repeated probing \rightarrow extend keystream to N
- Weaknesses in RC4 key scheduling algorithm
 - I are class of weak keys can break secret key

Classes of attacks

- Authentication forging
 - WEP encrypts challenge using client-chosen IV
 - Recovery of keystream for a given IV allows reuse of the IV for forging WEP authentication
 - Doesn't provide key, so can't join LAN
- Realtime decryption
 - IV reuse and probing → construct dictionary of IVs and keystreams
 - Enables decryption in real time
 - Storage: 1500 bytes of keystream for each IV;

Tools

- WEPCrack—breaks 802.11 keys
 - http://wepcrack.sourceforge.net/
- AirSnort—breaks 802.11 keys
 - Needs only 5-10 million packets
 - http://airsnort.shmoo.com/
- NetStumbler—access point reconnaissance
 - http://www.netstumbler.com

WEP suckage

- Same key reused over and over again
 Per-packet IV isn't enough
- Need to increase keyspace an attacker must analyze
 - Generate new keys (not just IVs) periodically
 - Use unique per-client keys
 - These are our first requirements...

Other problems

- Rogue access points
 - Mutual authentication—AP authenticates to client
- Disassociation attacks
 - Assoc/disassoc messages are unencrypted and unauthenticated
 - Fix with keyed message integrity check
- Unauthorized use or monitoring
 - Incorporate user and computer authentication

802.1x



Solution today: 802.1x

- Port-based access control mechanism defined by IEEE
 - Works on anything, wired and wireless
 - Access point must support 802.1x
 - No special WIC requirements
- Allows choice of authentication methods using EAP
 - Chosen by peers at authentication time
 - Access point doesn't care about EAP methods
- Manages keys automagically
 - No need to preprogram WICs

Is 802.1x enough?

- No
- It does solve:
 - Key discovery by changing keys often and using different keys for each client
 - Rogue APs and man-in-the-middle attacks by performing mutual device authentication
 - Unauthorized access by authenticating users and computers
- It does not solve:
 - Packet and disassociation spoofing because 802.1x doesn't use a keyed MIC

Clarifying terminology

- 802.11 is the specification for over-the-air wireless networks
- 802.1x is a PHY-independent specification for port-based access control
- Combining them makes sense
- There is no such thing as 802.11x
 - But there is work on something called 802.11i

802.1x over 802.11



Association and authentication The 802.11 association happens first

- Need to talk to the AP and get an IP address
- Open authentication—we don't have the WEP key yet
- Access beyond AP prohibited until authN succeeds
 - AP drops non-EAPOL traffic
 - After key is sent in EAPOW-key, access beyond AP is allowed
- Security conversation between supplicant and authentication server
 - Wireless NIC and AP are passthrough devices

Before authentication



After authentication



802.11/802.1x state machine



Encryption keys

- Client and RADIUS server generate peruser session WEP keys
 - Never sent over the air
 - RADIUS server sends key to AP (encrypted with RADIUS shared secret)
- Access point has a global WEP key
 - Used during AP authentication to client
 - Sent in EAPOW-key message
 - Encrypted with session key
- Session keys regenerated when...
 - Key time exceeded (60 minute default)
 - Client roams to new AP

Extensible authentication protocol



EAP

Link-layer security framework

- Simple encapsulation protocol for authentication mechanisms
- Runs over any link layer, lossy or lossless
- No built-in security
 - Doesn't assume physically secure link
 - Authentication methods must incorporate their own security

Authentication methods

- EAP allows choice of authentication methods
- For mutual authentication—
 - TLS: authentication server supplies certificate
 - IKE: server demonstrates possession of preshared key or private key (certificate)
 - Kerberos: server demonstrates knowledge of session key

AuthN supported in Windows

EAP-MD5 disallowed for wireless

- Can't create encrypted session between supplicant and authenticator
- Would transfer password hashes in the clear
- Cannot perform mutual authentication
 - Vulnerable to man-in-the-middle attacks
- EAP-TLS in Windows XP release
 - Requires client certificates
 - Best to have machine and user
- Service pack 1 adds protected EAP

Protected EAP (PEAP)

- Extension to EAP
- Allows use of any secure authentication mechanism for EAP
 - No need to write individual EAP-enabled methods
- Windows PEAP allows:
 - MS-CHAPv2—passwords
 - TLS—certificates
 - SecurID
- For many deployments, passwords still (alas) are necessary

EAP architecture





PPP	802.3	802.5	802.11	Anything	media laver
					layor

Note

- Do not configure IAS and XP for both—
 - EAP-TLS alone
 - PEAP with any method
- Man-in-the-middle vulnerability
- If you need TLS and MS-CHAPv2 together—
 - Deploy only PEAP
 - Select both MS-CHAPv2 and TLS methods

How it works: The Windows logon process over PEAP with MS-CHAPv2



Security requirements, again

- Mutual device authentication
 - Workstation and AP
 - No rogue access points
 - Prevents man-in-the-middle attacks
 - Ensures key is transferred to correct entity
- User authentication
 - No unauthorized access or interception
- WEP key uniqueness and regeneration
- Stop packet/disassociation spoofing

Windows domain logon

- Two logons occur
 - Machine
 - User
- Machine accounts look like user accounts
 - Certificate credential
 - User ID/password/domain credential
 - Take advantage of this

Windows PEAP authentication First phase—machine logon

- 802.11 association
- Authenticate AP
- Authenticate computer
- Transition controlled port status
 - For machine account access to authorized resources
- Second phase—user logon
 - Authenticate user
 - Transition controlled port status
 - For user account access to authorized resources

Windows PEAP authentication First phase

- Supplicant performs regular 802.11 association
- Supplicant sets up TLS channel with authenticator and requests authentication server's certificate
- Supplicant—
 - Verifies name and dates on certificate
 - Validates chain

Our requirements so far

- Mutual device authentication
 - Workstation and AP
 - No rogue access points
- User authentication
 - No unauthorized access or interception
- WEP key uniqueness and regeneration
- Packet/disassociation spoofing

Windows PEAP authentication First phase

- Supplicant sends machine credentials to authenticator over previously-established TLS channel
 - Authenticator checks validity by contacting authentication server (RADIUS)
 - Authentication server contacts directory to verify credentials

Windows PEAP authentication First phase

If valid, RADIUS generates WEP key

Authenticator delivers key to supplicant and transitions controlled port status to permit supplicant access to LAN (to resources allowed access through machine account only)



Our requirements so far

- Mutual device authentication
 - Workstation and AP
 - No rogue access points
- User authentication
 - No unauthorized access or interception
- WEP key uniqueness and regeneration
- Packet/disassociation spoofing

Windows PEAP authentication Second phase

Logon dialog appears

- Supplicant sends user credentials to authenticator
- Authenticator checks validity by contacting authentication server (RADIUS)



Authentication server contacts directory

- If valid, authenticator extends controlled port status to permit supplicant *full* access to LAN
- User loaded on to domain

Our requirements so far

- Mutual device authentication
 - Workstation and AP
 - No rogue access points
- User authentication
 - No unauthorized access or interception
- WEP key uniqueness and regeneration
- Packet/disassociation spoofing

Why use machine accounts?

- Domain logon required for:
 - Machine group policies
 - Computer startup scripts
 - Software installation settings
- When user account passwords expire
 - Need associated WIC and transitioned controlled port for user notification and change dialog
 - Machine account logon phase allows password expiration notices and changes to occur normally
 - Cisco's LEAP can't deal with this
 - No facility for machine authentication

Why passwords?

- Not all customers are ready for a PKI
- Managing user certificates stored on computer hard drives will always be painful
 - Some personnel might roam among computers
 - Smartcards solve this
 - Technical and sociological issues can delay or prevent deployment
- PEAP enables (pretty) secure wireless now
 - Allows easy migration to certificates and smartcards later

Remaining vulnerabilities



Remaining vulnerabilities

- Two related vulnerabilities not addressed with 802.1x
 - Bit flipping with known IVs \rightarrow packet spoofing
 - Disassociation denials of service
- Simple addition to 802.1x will solve both

Bit-flipping attacks

- WEP doesn't perform per-packet authentication
 - IC is not a keyed message integrity check
 - Flipped bits in WEP packet → recalculated IC
- To spoof or replay:
 - Flip bits in WEP packet where IV is known
 - AP accepts packet
 - Layer 3 device rejects, sends predictable response
 - Build response database and derive key

Disassociation attacks

- 802.11 associate/disassociate messages are unauthenticated and unencrypted
- Attacker can forge disassociation message
 - Bothersome denials of service

Solution: keyed IC

- Change behavior of WEP's IC
- Derive key from seed value, source and destination MACs, payload
 - Any change to these will alter the IC
- Include in every WEP packet

Deployment



System requirements

- Client: Windows XP service pack 1
- Server: Windows Server 2003 IAS
 - Internet Authentication Service—our RADIUS server
 - Certificate on IAS computer
- Backporting to Windows 2000
 - Client and IAS must have SP3
 - No zero-config support in the client
 - See KB article 313664
 - Supports only TLS and MS-CHAPv2
 - Future EAP methods in XP and 2003 might not be

Setup

Build Windows Server 2003 IAS server

- Join to domain
- Enroll computer certificate
 - Register IAS in Active Directory
- Configure RADIUS logging
 - Add AP as RADIUS client



Configure AP for RADIUS and 802.1x



- Configure clients
 - Don't forget to import OA reat \mathbf{A}

Access policy

- Policy condition
 - NAS-port-type = Wireless IEEE 802.11 and Wireless other
 - Windows-group = <some group in AD>
 - Optional; allows administrative control
 - Should contain user and computer accounts
- Profile
 - No regular authentication methods
 - EAP type: protected EAP; use certificate from step 3
 - Encryption: only strongest (MPPE 128-bit)
 - Attributes: Ignore-user-dialin-properties = True

What else?



Interoperability

- PEAP standards authors
 - Microsoft
 - Cisco
 - RSA
- Our implementation is version 0
 - Not compatible with version 1
- Working towards interoperability
 - PEAP allows servers and clients to support multiple versions

802.1x alternative

- WPA (Wi-Fi protected access)
- Includes TKIP (temporal key integrity protection)
 - Uses RC4, rotates keys every 10,000 packets
 - Combines shared 128-bit key with client MAC and 128-bit IV
 - Provides key uniqueness
- WPA relies on 802.1x for user and mutual device authentication
- In beta now for Windows XP

The future—long term

IEEE is working on 802.11i

- Replacement for WEP
- Includes TKIP, 802.1x, and keyed IC
- Uses AES
- Addresses all currently known vulnerabilities and poor implementation decisions
- Need to be IEEE member to read work in progress
- Expected ratification in September 2003

References

- Security of the WEP Algorithm
 - http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html
- 802.1x--Port Based Network Access Control
 - http://www.ieee802.org/1/pages/802.1x.html
- PPP Extensible Authentication Protocol
 - http://www.ietf.org/rfc/rfc2284.txt
- PPP EAP-TLS Authentication Protocol
 - http://www.ietf.org/rfc/rfc2176.txt
- Protected EAP Protocol
 - ftp://ftp.rfc-editor.org/in-notes/internet-drafts/draft-



© 2003 Microsoft Corporation. All rights reserved. This presentation is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY.