Weaknesses in LEAP Challenge/Response

Joshua Wright

Joshua.Wright@jwu.edu

LEAP == Cisco Marketshare

- LEAP is Cisco's plan to have controlling marketshare in the 802.11 AP product space
- Lightweight Extensible Authentication Protocol
  - Also known as Cisco EAP
  - Easy to install and configure
  - Easy to support (unified supplicant)
  - It must be secure, right?

LEAP is a closed EAP type

- LEAP specification only opened to business partners under NDA
- Client is licensed to other NIC manufacturers (D-Link, SMC, 3Com, Apple)
- AP code to do LEAP is IP of Cisco
- Information gathered here is collected from:
  - Packet captures of LEAP transactions
  - http://lists.cistron.nl/pipermail/cistron-radius/2001-September/002042.html

LEAP makes the world safer

- Provides authentication and data privatization
  - Uses modified MS-CHAPv2 challenge/response in the clear
  - Uses mutual authentication to mitigate MITM attacks
  - Uses short-lived WEP keys to encrypt data
  - Prevents usage of weak IV's from the AP

MS-CHAPv2 Weaknesses

- MS-CHAPv2 weaknesses apply to the LEAP exchange
  - No salt in stored NT hashes
    - Permits pre-computed dictionary attacks
  - Weak DES key selection for challenge/response
    - Permits recovery of 2 bytes of the NT hash
  - Username sent in clear-text
  - We can deduce authentication passwords

LEAP STA Challenge/Resp

1. AP issues random 8-byte challenge to STA
2. STA uses 16 byte NT hash (MD4) of password to generate 3 DES keys
   1. $NT_1 – NT_7$
   2. $NT_8 – NT_{14}$
   3. $NT_{15} – NT_{16}$ + "\0 \0 \0 \0 \0"
3. Each DES key is used to encrypt the challenge (each generating 8 bytes of output)
4. STA sends 24-byte response to challenge
5. AP issues success or failure message

Response leaks 2 bytes of NT hash

- The third DES key is weak
  - Five NULL's are consistent in every challenge/response
  - Leaves only 2^16 possibilities
    - Can calculate 2^16 DES with a known challenge in < 1 sec
- Significantly reduces search space
  - Known hash bytes significantly reduces hash possibilities
  - `'$ grep "B1B2$" nthash-dict > possible-passwords'`
  - From 2.5 million passwords, usually leaves ~30

Our Attack

- Take a large password list, calculate MD4 hashes to generate a password+NT hash list
- Capture LEAP challenge/response
  - Extract username, challenge, response
  - Calculate the last 2 bytes of the NT hash from the response
- Search through pass+hash list for hashes with matching bytes
- Use matching entries to encrypt the challenge
  - Matching captured and calculated response will indicate the user's password

Implementation – asleap-imp

- genkeys
  - Accepts a dictionary list of passwords and generates a "password \t hash" output file
- asleap
  - Reads from a pcap file, or from a network interface in RFMON mode
  - Watches for LEAP challenge/response
  - Calculates last two bytes of NT hash
  - Searches through genkeys output file for matches
  - Reports the user password

asleap-imp Features

- Search mode
  - Hops on all channels with user-specified hopping duration
- Active mode
  - Identifies active STA's
  - Injects spoofed frame sending LEAP Logoff, followed by a deauthenticate frame to the STA
    - Forces the victim to participate in a new challenge/response
- Saves LEAP exchange in a pcap file for later analysis
  - Hack from another machine with more disk space/larger genkeys password list

asleap-imp Demo