1 ARProtokoll Schwächen
- Felix Gröbert 2004 - Version 1.0
- http://groebert.org/felix/

2 Vortrags Struktur
- Protokoll
- Angriffstechniken
- Angriffsszenarios
- Programme
- Resultate
- Demo
-

3 Motivation?
- 70% der Angriffe von ‘innen’
- jeder 500ste ‘hasst’ seine Firma
- ARP Schwächen oft unterschätzt

4 Address Resolution Protocol
- Protokoll zum Zuordnen von Hardware- zu IP Adressen

5 RFC
- ARP ist definiert im RFC 826 von 1982
- Generieren von Protokoll Adressen aus der Hardware Adresse
- Protokolle wie DOD TCP, Xerox BSP, DECnet

6 mehr RFC
- RFC 826 wird durch RFC 903, 1293, 1735, 2390 ergänzt (Inverse / Reverse ARP)
- Zeroconf - in Arbeit bei IETF

-

7 Einordnung OSI-Model

8 meine Einordnung

9 Wie gehts?
- 192.168.23.3 möchte via IP mit 192.168.23.1 sprechen, Gesucht: MAC
- arp who-has 192.168.23.1 tell 192.168.23.3
- arp reply 192.168.23.1 is-at 00:0f:66:d3:fc:0n
- 192.168.23.3 weiss jetzt die MAC von 192.168.23.1

10 MAC (1)
- Media Access Control
- Ziel: eindeutige Identifizierung eines Netzwerkdevice
- 48 Bit = 6 Bytes = 6 hexadezimale Zahlen
- Erste 3 Bytes Herstellerkennung
- ac:de:48 ist der Hersteller “Privat”

11 MAC (2)
- Herstellerpräfixe im Netz können Rückschlüsse geben
- ifconfig eth0 ether 01:01:ac:1d:b4:be
- Broadcast ff:ff:ff:ff:ff:ff
- Multicast 01:00:5e:xx:xx:xx

12 ARP cache
- MAC-IP Zuordnungen müssen gecacht werden
- cache muss sich aktuell halten (timers)
- requests-src Daten und reply Daten kommen in den Cache

13 Ethernet Paket

14 ARP Paket Aufbau

15 Paketbündelparameter

- Paket Parameter:
  - Opcode (request / reply)
  - Ethernet DST
  - Ethernet SRC
  - ARP SRC MAC
  - ARP SRC IP
  - ARP DST MAC
  - ARP DSP IP

16 Standard request

17 Standard reply

18 bessere switche

- switch = besseres hub
- VLANs (viel segmentierung)
- switchport protected (Cisco)
- dynamic ARP inspection (Catalyst)

19 Angriffstechniken

- MAC spoofing
- ARP spoofing
- ARP cache poisoning

20 MAC spoofing

- O will Traffic lesen der an B geht
- O sendet von seinem Port ein beliebiges Paket mit

- ETH DST(O)
- ETH SRC(B)

21 Wirkung

- Der Switch hat seine Tabelle geändert, Pakete für MAC(B) gehen nun an Port 3
- Nachteile
  - O muss schnell sein
  - B wird unereichbar
  - Pakete können durchfallen
  - Switch Admin
-

22 ARP spoofing

- Angreifer antwortet schneller auf request als das request-Ziel
- ARP cache übernimmt (meist) ersten Reply

23 Nachteile

- Nur bereits im ARP Cache existierende IP Adressen werden manipuliert
- Wettrennen gegen echten reply

24 ARP cache poison

- optimale Angriffstechnik
  - eintragen oder ändern von MAC/IP Zuordnungen im cache des Opfer
  - legales Paket auf Layer 2
  - von den meisten Switchen nicht erkannt

25 Eintragen

- eintragen

26 Wirkung

- Angreifer täuscht mit der IP Adresse des Gateway einen request vor
- Opfer legt aus Effizienzgründen die source Daten des Angreifer im Cache ab
- Wenn jetzt eine Verbindung zum oder über den Gateway geht, läuft diese erst beim Angreifer auf (Layer 2)

27 Ändern

28 Wirkung

- replys werden ohnen vorhergegangen request akzeptiert
- der reply ist legal
- aus Effizienz wird Angreifer Gateway

29 Non-Operating Systems

- Windows XP, 2000, 98
- Mac OS X
- Linux 2.6

30 Operating Systems

- FreeBSD
- Nov 12 06:18:41 fb /kernel: arp: 24.237.82.161 moved from 00:40:c7:81:22:04 to 00:04:ac:1a:4e:e7 on dc0
- OpenBSD
- Feb 7 12:03:42 ob /bsd: arp info overwritten for 66.68.195.49 by 00:30:7b:ff:50:70 on xl0

31 Angriffsszenarios

- Denial of Service
- Firewall escaping
- Sniffing
- Proxying, Hijacking, Man in the Middle

32 denial of service

- viel zu viele Möglichkeiten
- broadcast, timeouts, ddos

33 firewall escaping

- Gateway vergiften, Angreifer bekommt Opfer IP
- Opfer vergiften, Angreifer wird Gateway
- Angreifer sortiert Opfer traffic von seinem traffic

34 Sniffing

- vergiften des cache der Opfer
- ip_forward 1
- alles läuft durch Angreifer
-

35 MITM

- Angreifer host in der Mitte
- Beide hosts vergiften (sorgsam)
- Einen host vergiften (ausreichend)

36 Programme

37 arp-sk

- SwissarmyKnife für ARP
- Kommandozeileninterface
- libnet
- sehr flexibel, wie netcat

38 arp-sk

39 dsniff

- Kommandozeileninterface

- verschiedene tools
- passiv: sniffen
- aktiv: mitm
- libpcap, libnet, libnids

40 dsniff
- dsniff (password sniffer)
- filesnarf, mailsnarf, urlsnarf
- arpspoof, macof (besser: arp-sk)
- dnsspoof
- sshmitm (SSHv1), webmitm (HTTPS)

41 ettercap-ng

42 ng
- GTK, ncurses, relativ bunt
- sniffing, mitm, deciphering, pen testing, bla
- einfach erweiterbar mit plugins

43 Resultate
- Datenverkehr kann in normalen LANs beliebig abgehört und manipuliert werden
- unter Umständen auch SSH oder HTTPS
- der normale User vertraut seiner Netzwerkanbindung
- durch gepatchte binarys oder client-side vulnerabilities können hosts kompromittiert werden
- wegen ARProtokoll Schwachstellen

44 MITM Code Projekt
- Opfer geht zu einer HTTP-download Seite
- Angreifer übernimmt auf Ethernet Ebene Gateway
- angefordertes binary wird mit payload gepatcht

- CIH ‘98 keine Änderung der exe Größe
- ELF angeblich noch einfacher

45 demo: LAN

- 10.0.0.3 Angreifer - Gentoo Linux
- 10.0.0.2 Opfer - Windows XP
- 10.0.0.1 Server - Mac OS X

46 Ressourcen

- Folien, Links, Referenzen, Software
- http://arp.infoflood.de

47 DON’T PANIC

- static router MAC
- VLANs
- honeynets, NIDS, IDS
- TTL analyse und sha-1 hashes
- HTTPS root-CAs