Taranis: Invisible Traffic Redirection on Ethernet Switches

Jonathan Wilkins

jwilkins@bitland.net

http://www.bitland.net/taranis

Introduction

- We'll cover:
  - Ethernet ARP Cache Corruption (briefly as background)
  - Ethernet Switch CAM Table Poisoning
  - Defenses
- Expected background
  - You know what a sniffer is
  - You are familiar with network traffic analysis and basic network protocols

OSI Model

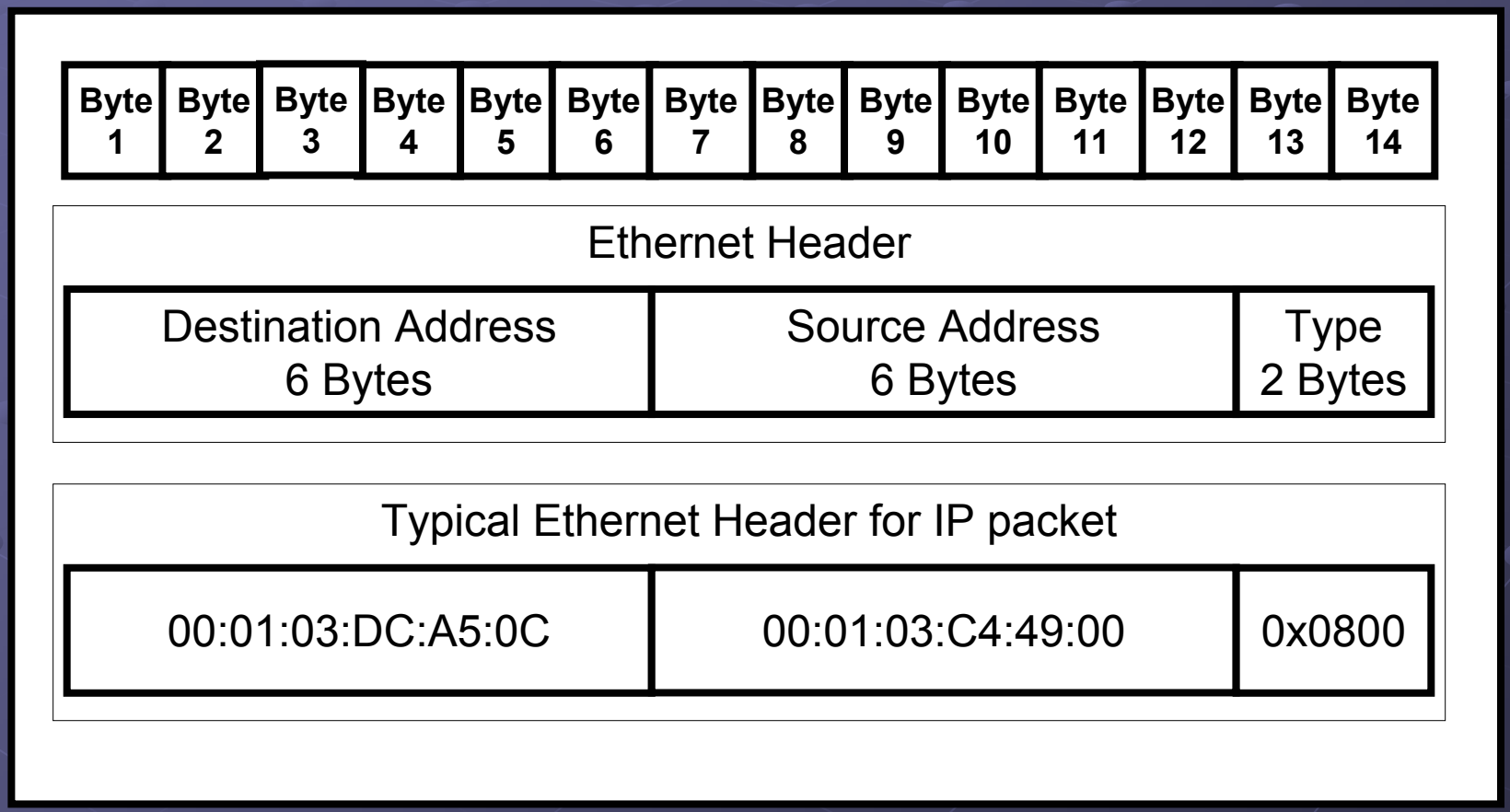
- Application – DNS, SNMP, HTTP
- Presentation – SSL
- Session –
- Transport – TCP, SPX, NetBIOS
- Network – IP, IPX, ARP, Routers
- Data Link – Ethernet, Bridges, Switches
- Physical – Cabling, Hubs, Repeaters

How Ethernet Works

Frame Layouts and ARP

Ethernet

- Data Link layer



Ethernet Address Resolution

- For machines to be able to communicate with each other at all, they each need to have a list of IP address to Hardware address mappings
- For these machines to be able to communicate in an ad-hoc fashion, they need to have some way to query for IP address to Hardware address mappings.
- This is done through ARP (Address Resolution Protocol)

ARP

- Normally, when a host needs to talk to another host, it checks it's ARP cache to see if it already has an entry for the other host. If it does, just uses it.
- If not, it sends out an ARP query.
- Gratuitous ARP

ARP Request Format

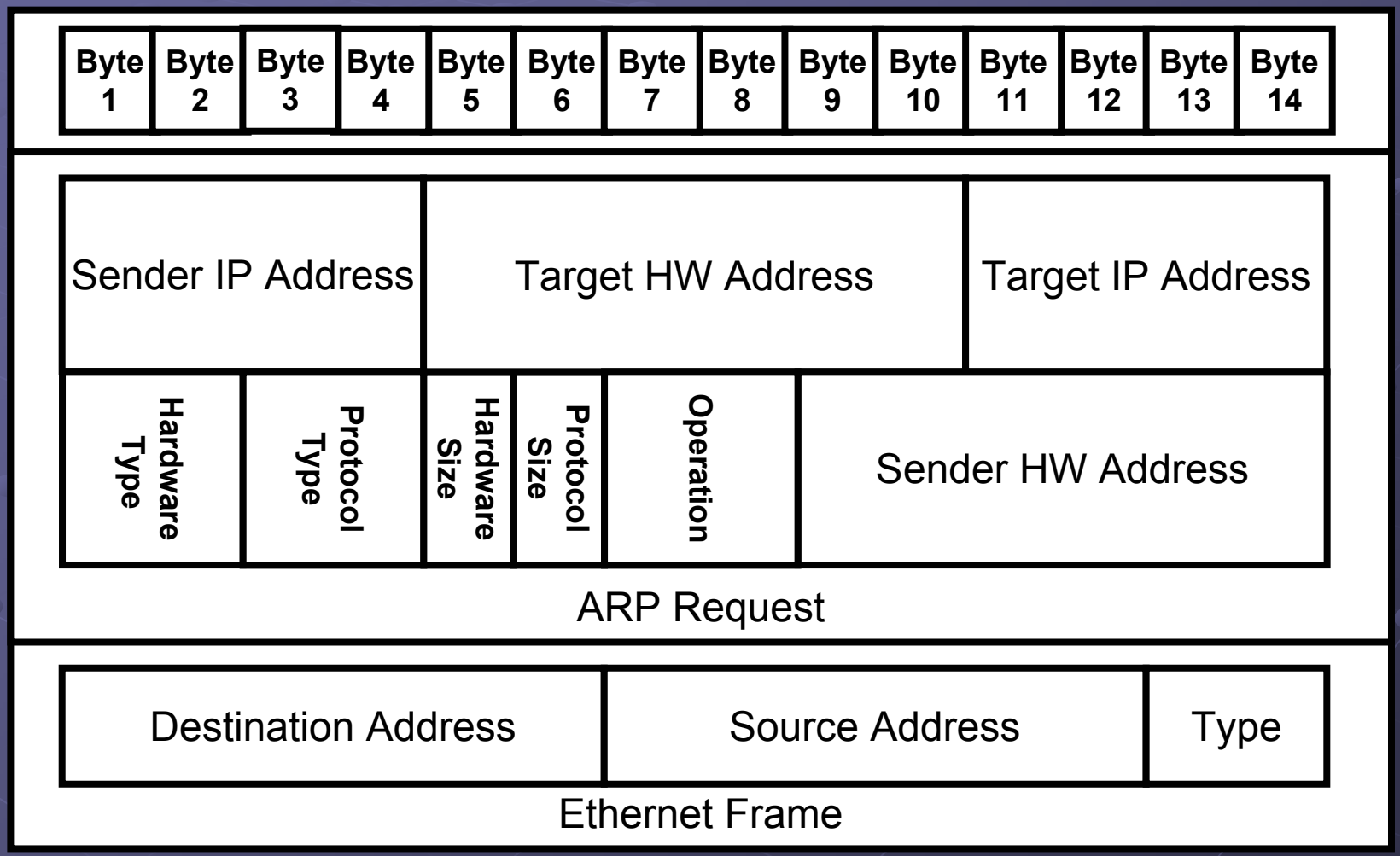
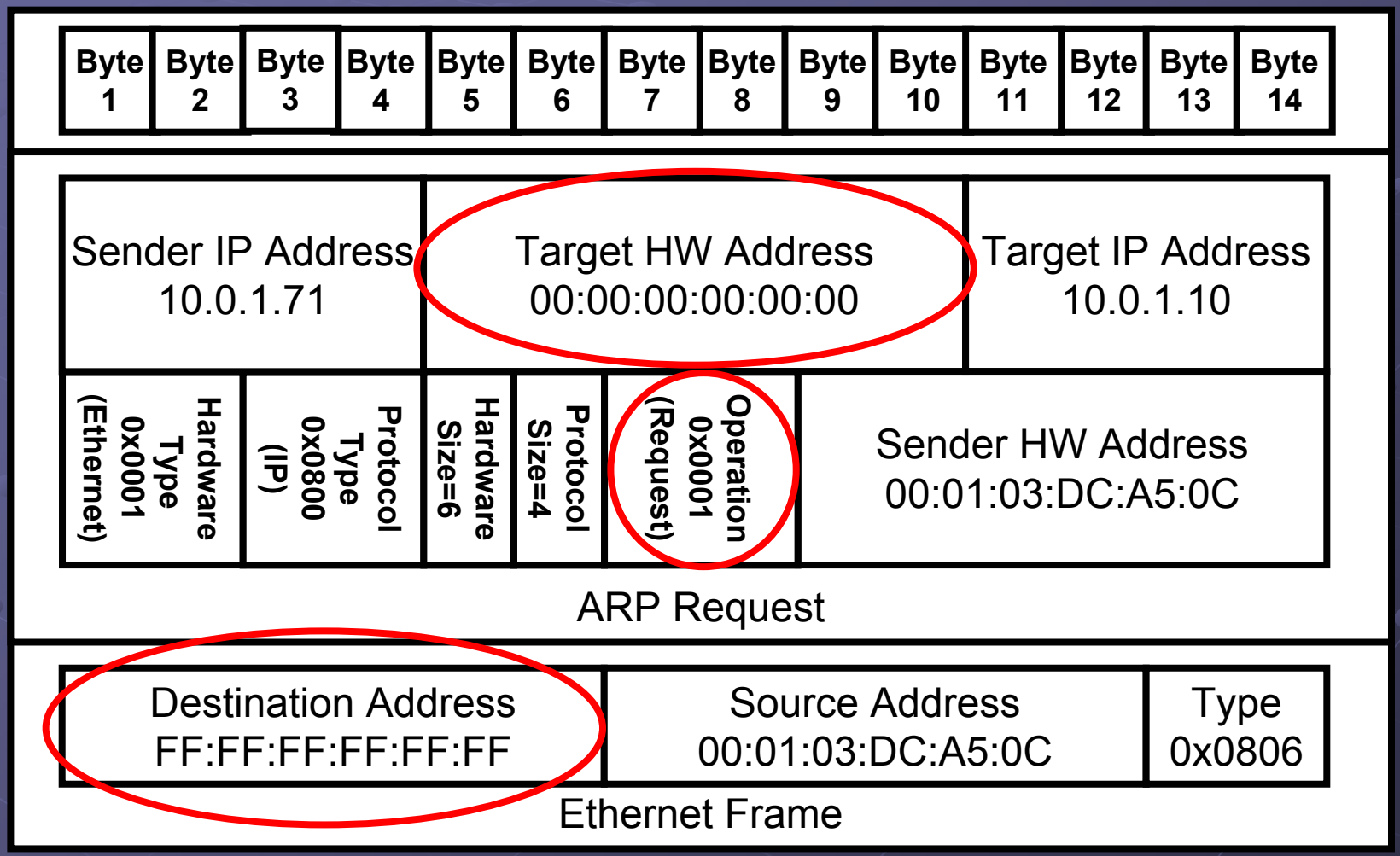
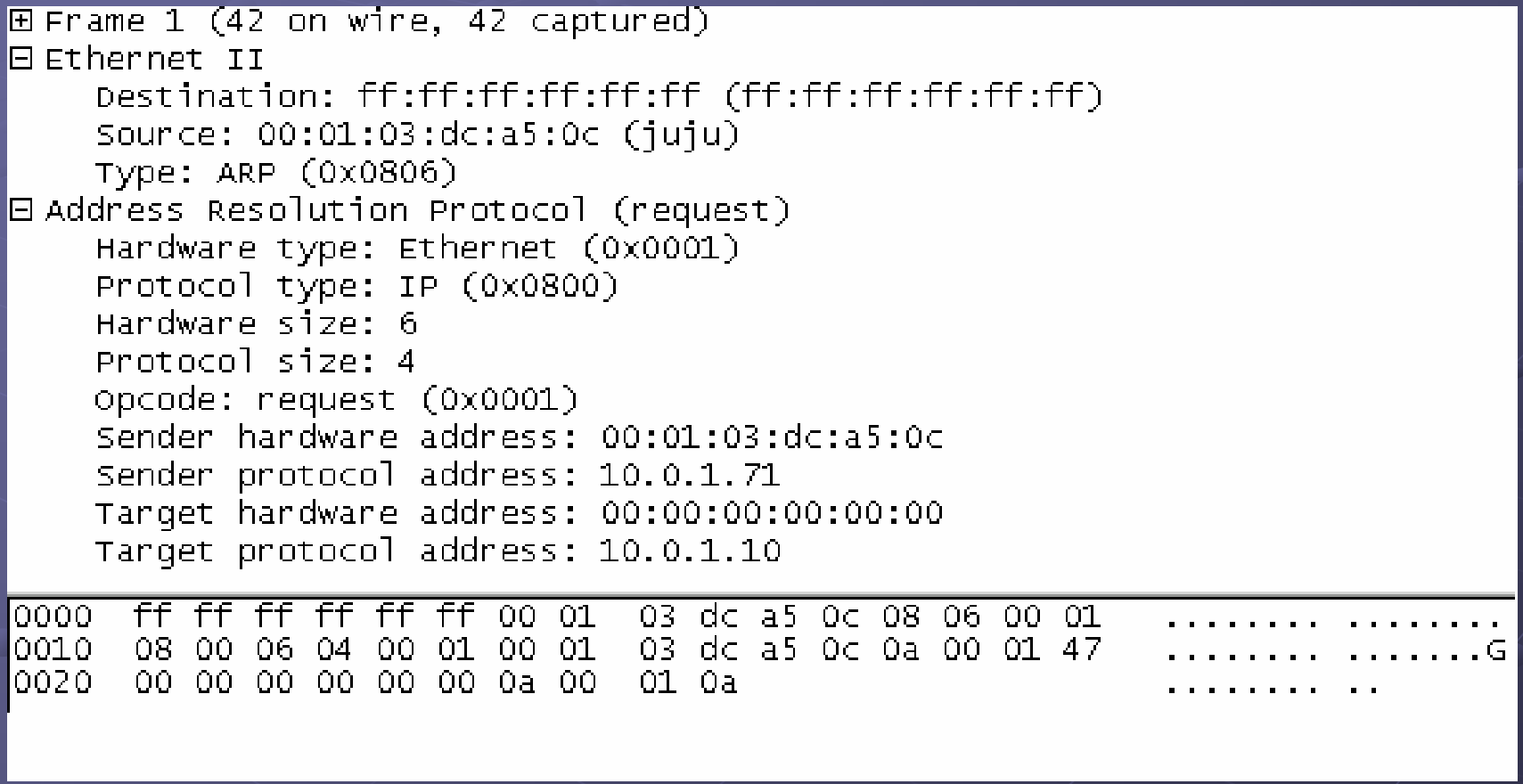


Diagram of ARP Request Packet



Ethereal Display of ARP Request



ARP Reply Format

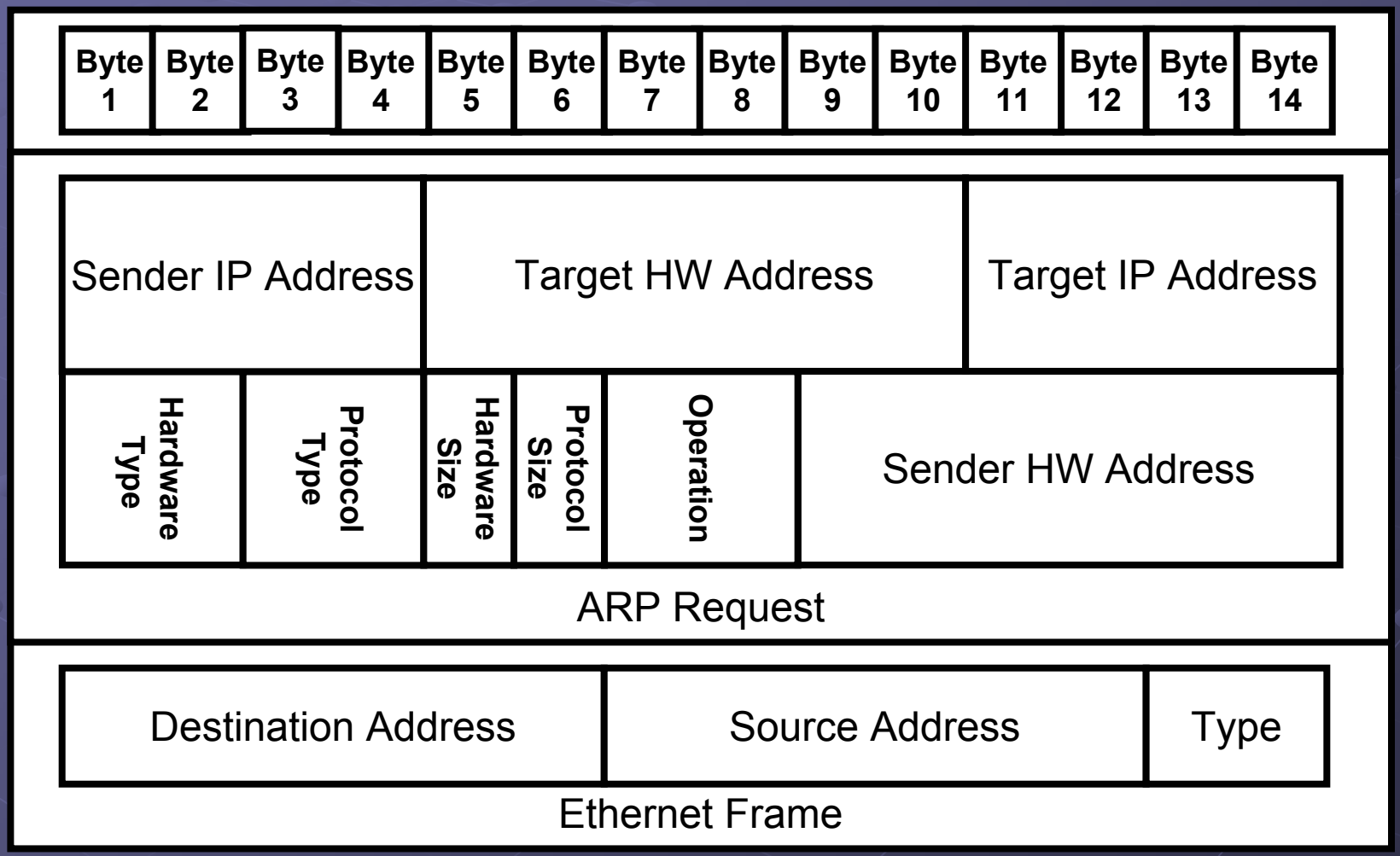
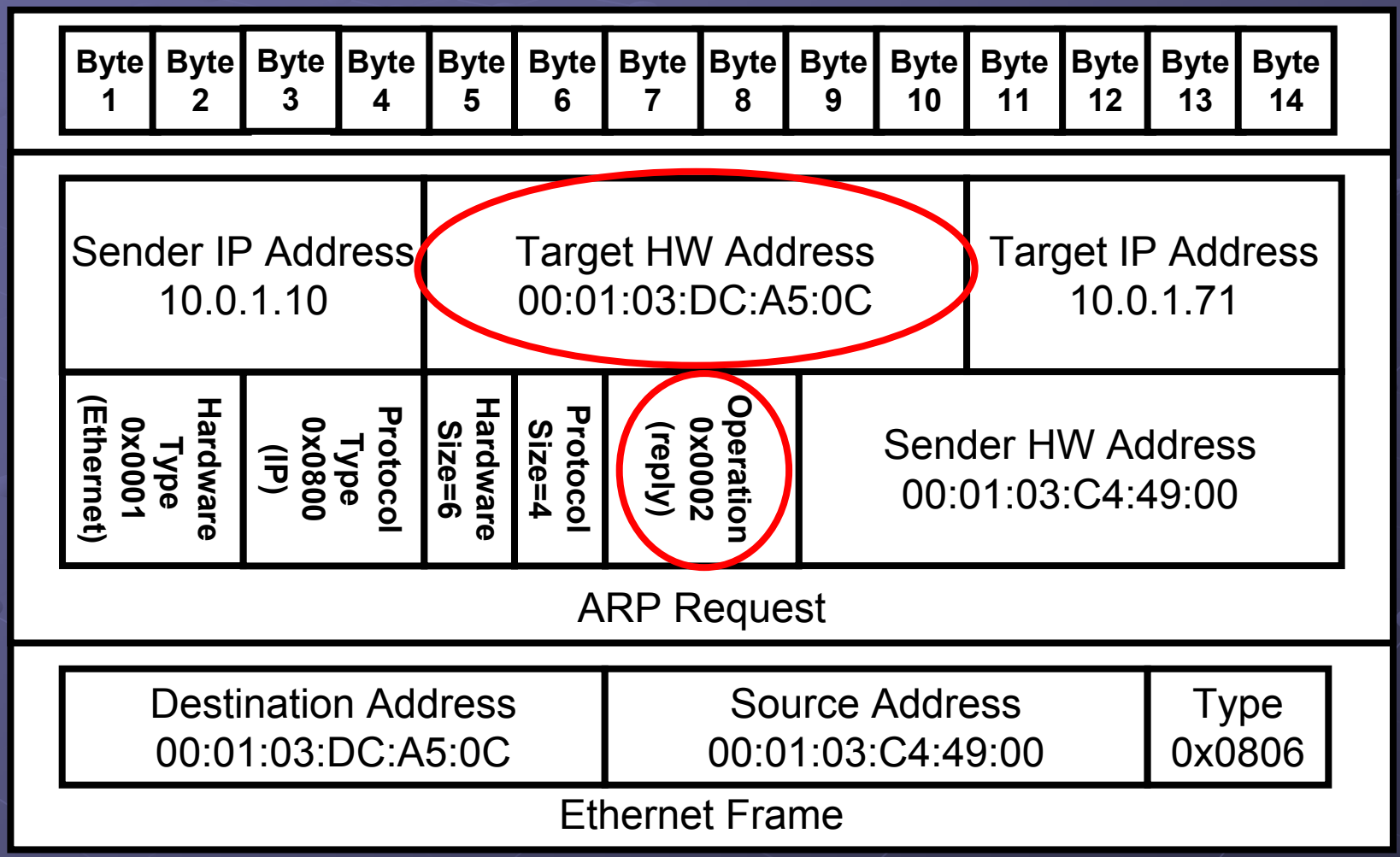


Diagram of ARP Reply Packet



Ethereal Display of ARP Reply

```
⊞ Frame 2 (60 on wire, 60 captured)
⊟ Ethernet II
    Destination: 00:01:03:dc:a5:0c (juju)
    Source: 00:01:03:c4:49:00 (3COM_c4:49:00)
    Type: ARP (0x0806)
    Trailer: 000000000000000000000000000000000...
⊟ Address Resolution Protocol (reply)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (0x0002)
    Sender hardware address: 00:01:03:c4:49:00
    Sender protocol address: 10.0.1.10
    Target hardware address: 00:01:03:dc:a5:0c
    Target protocol address: 10.0.1.71
```

```
0000  00 01 03 dc a5 0c 00 01  03 c4 49 00 08 06 00 01   ........ ..I.....
0010  08 00 06 04 00 02 00 01  03 c4 49 00 0a 00 01 0a   ........ ..I.....
0020  00 01 03 dc a5 0c 0a 00  01 47 00 00 00 00 00 00   ........ .G......
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....
```

The Role of Switches

- Switches increase performance by creating virtual circuits
  - Aggregate bandwidth is boosted
- Switches don't add security
- A VLAN is functionally a separate switch. Feel free to interchange the words Switch and VLAN

ARP Cache Corruption

- When a given host sends out an ARP request for a host's IP address, (the network gateway is the usual target) the attacking host replies to it, using it's own hardware address.
- Traffic then goes through the attacking host, thereby giving it access to all kinds of interesting information.
- This doesn't disrupt normal traffic, as the attacking host forwards all traffic to the correct gateway host.

Detecting ARP Cache Corruption Attacks

- Presuming that an administrator has access to a monitoring port on the switch, detection is easy, just check for multiple MAC addresses responding to a given IP address.
- ARPWatch can do this through sniffing or by SNMP
  - (ftp://ftp.ee.lbl.gov/arpwatch.tar.gz)

A New Attack - Reasoning

- Switches can auto-configure themselves
- Switches can either have a single host or another switch/hub on each port
- When a device is moved from one port to another, nothing needs to be done to the switch.
- Therefore: The switch must automatically maintain a list of Ethernet address to port mappings
- Question: Can we trick the switch into sending traffic to the wrong port by tricking it's auto-configuration logic?

A New Attack - Reasoning

- What attacks are likely?
  - Send a packet with the victim's Ethernet address and see if it will start sending that data to both ports (Limited Hub Mode)
  - Send a packet with the victim's Ethernet address and see if it will redirect traffic
  - See if we can resend the traffic and have it picked up by the victim for a full MITM attack
  - Send packets with source addresses of other hosts on the network on our port and try and get the switch to drop into hub mode (a la macof)

Results

- We **weren't** able to get any tested switch to send to multiple ports
- We **weren't** able to get any tested switch to drop to hub mode
- We **were** able to get traffic redirected to a different port on all tested switches
- We **weren't** able to get a simple MITM attack functioning, but we have some ideas that may be sufficient to extend the attack
- We **were** able to generate an easy DOS on all tested switches

Why it works

- Switches attempt to over simplify the complicated affair of configuration
- Only expensive managed switches allow port locking and even there, the administrative cost of doing so is prohibitive

What is the CAM Table?

- Basically a really efficient lookup table
- Present on all modern switches
- CAM == Content Addressable Memory
- For more information on the CAM table and how it is updated check out http://routergod.com/gilliananderson or http://www.isdmag.com/editorial/1998/systemdesign9801.html

What is the CAM Table?

- This internal table looks something like this:

Port	Ethernet Addresses	Host or Uplink
1	01:00:af:34:53:62	Single host
2	01:e4:5f:2a:63:35 00:c1:24:ee:62:66 ...	Switch or Hub
3	11:af:5a:69:08:63 00:17:72:e1:72:70 ...	Switch or Hub
4	00:14:62:74:23:5a	Single host

Switch Behavior

- 3 update behaviors were seen
  - Last packet updates CAM table
  - Timeout before CAM table update
  - Port is locked
- Each approach has problems
  - Last packet updates – Trivial to redirect traffic
  - Timeout – Once port is redirected, keeping control is easy. Requires less traffic to keep control
  - Lock – Huge administrative burden associated

Results – Switch Update Behavior

Cisco 2900 MXL and Cisco 2900 XL (Port Security Off)	Timeout (~30 seconds)
Cisco 2900 MXL and Cisco 2900 XL (Port Security On)	Traffic redirected only on reboot or cable disconnect
LinkSys EZXS88W	Last Packet Updates
HP ProCurve 2424M	Last Packet Updates
3Com SuperStack II/1100	Last Packet Updates

Results – Why it's Invisible

- There's no way to tell which port traffic is coming from at the Ethernet level
- All unmanaged switches are therefore totally vulnerable
- There are no tools to track the MAC address to Port mappings for the few managed switches that have a way to query these mappings

Results - Why It's Invisible

- Even if an administrator attempted to poll the switch continually, this attack could be slipped between queries
- Only does protocol specific traffic. Other hosts have no way to tell that there has been a substitution

ARP Corruption Versus CAM Table Poisoning

- Layer 3
- ARP provides a good Man In The Middle attack
- Tools exist to detect this attack. The attack consists of packets with a definite format

- Layer 2
- No good Man In The Middle attack
- Impossible to generate a single signature for the attack

Taranis

- Proof of concept code
- Taranis spawns two threads
  - First thread loops, spoofing packets with an Ethernet source address of the target machine – This tricks the switch into redirecting traffic
  - Second thread pretends to be a POP or IMAP server
- Any connection attempts to port 110 (POP) or 143 (IMAP) are handled by Taranis

POP3

- A normal POP login sequence looks like this:

```
[jwilkins@toxygene ~]$ nc localhost 110
+OK QPOP (version 2.53) at mail.net starting.  <39898.1010976721@mail.net>
USER jwilkins
+OK Password required for jwilkins.
PASS mypasswd
+OK jwilkins has 592 messages (7597672 octets).
QUIT
+OK Pop server at mail.net signing off.
```

IMAP

- A normal IMAP login sequence looks like this:

```
[jwilkins@toxygene ~]$ nc localhost 143
* OK mail.net IMAP4rev1 v12.264 server ready
A00001 LOGIN "jwilkins" “mypasswd"
A00001 OK LOGIN completed
A00002 LOGOUT
* BYE mail.net IMAP4rev1 server terminating connection
A00002 OK LOGOUT completed
```

Taranis

- For POP:
  - When a SYN is seen, a SYN/ACK is sent
  - When an ACK is seen, the POP banner is sent
  - Expects USER <username>\r\n
  - Sends “OK”
  - Expects PASS <password>\r\n
  - Resets the connection

Taranis

- For IMAP (even easier):
  - When a SYN is seen, a SYN/ACK is sent
  - When an ACK is seen, the IMAP banner is sent
  - Expects A0001 LOGIN “user” “password”\r\n
  - Resets the connection

Taranis

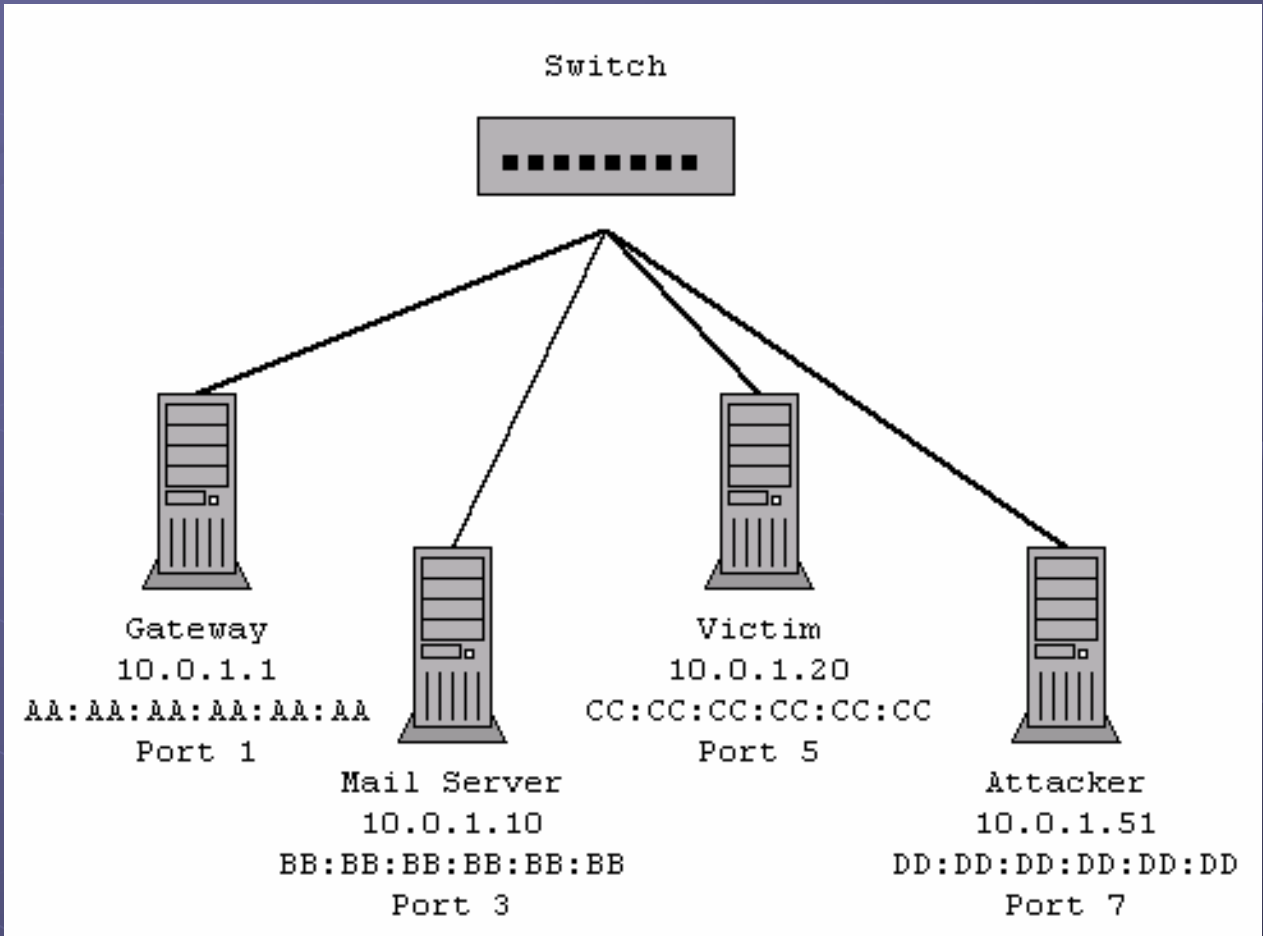
Taranis - POP3

Client	Server
SYN	SYN/ACK
ACK	+OK QPOP (version 2.53) at mail.net starting. <28815.1011055535@mail.net>\r\n
USER foo\r\n	OK\r\n
PASS bar\r\n	RST

Taranis - IMAP

Client	Server
SYN	SYN/ACK
ACK	* OK mail.net IMAP4rev1 v12.264 server ready\r\n
A00001 LOGIN “user” “pass”\r\n	RST

Taranis - Demo



- Break for demonstration of Taranis

Taranis – Future Directions

- Proof of concept code
- Any other protocol could be spoofed in this way
- A replacement log host seems like a fun idea
- An attacker could even just reconfigure a complete system with the target server's IP address and Ethernet MAC address
- There doesn't seem to be a way to perform a perfect man in the middle (MITM) attack with this technique
- It is, however, possible to queue packets and perform a protocol independent attack

Defences

- This attack can be almost invisible.
  - Ethernet doesn't indicate what port a given frame came across
- A generalized detection heuristic is difficult. Depending on the type of switch, the attack can be successful with just one properly timed packet. Cryptography is the only real defense.

Defences

- Do all clear-text logins via SSH or stunnel
  - This isn't perfect.. Man in the middle attacks are possible since users still don't have any clue about security and may accept a bad certificate or key
  - See Dug Song's WebMITM and SSHMITM at http://www.monkey.org/~dugsong/dsniff
- System must provide strong authentication
  - Use SSH public keys or Stunnel with certificates

SSH Tunnels

- ssh -L 110:mail.net:110 -l username -N mail.net
- Point your mail client at localhost instead of mail.net
- SSH will prompt you for a password
- You can use ssh-agent to simplify this
- SSH is available for Windows
- This does require shell access to the remote host. Obviously, if shell access isn't required for any other reasons, stunnel would be a better solution

Stunnel

- Full information on stunnel and certificates is available at

  http://www.stunnel.org/examples/client_cert.html

- On the client
  - stunnel –c –p mailnet.pem –d pop3 –r mail.net:pop3s
- On the server
  - stunnel –p /path/to/stunnel.pem –d pop3s –r pop3
- Stunnel is available for Windows

Summary

- Ethernet does not provide a private channel. It wasn't designed for this.
- All sorts of protocols that are common on intranets use clear text authentication (POP3, IMAP, SNMP, Telnet, FTP, Log servers, CVS)
- If an attacker gets intranet access, you've probably lost unless you have strong authentication and good crypto.
- Now that we're all aware of the need for protecting authentication details, setting up encrypted tunnels is a necessity

Summary

- Taranis allows
  - Invisible traffic redirection
    - No general heuristic possible for this attack
    - Polling the CAM table **may** work sometimes
  - Invisible Denial Of Service
    - Rendering switches less secure than hubs in certain situations
  - Cryptography is the best protection
  - Intranets shouldn’t be trusted

References

- Taranis
  - http://www.bitland.net/taranis
  - http://www.bitland.net/taranis/taranis.[pdf,ppt]
- Setting up encrypted tunnels
  - SSH (http://www.openssh.com)
    - http://www.employees.org/~satch/ssh/faq/
  - Stunnel (http://www.stunnel.org)
    - http://www.freebsddiary.org/stunnel.php
- Ethereal (http://ethereal.zing.org)
- Ethernet Vendor codes (http://www.cavebear.com/CaveBear/Ethernet/vendor.html)

Thanks

- Jesse jesse@bitland.net for the original idea
- Skyper skyper@segfault.net for all the help testing switches
- Ed ed@apache.org help testing switches