CS 155 Spring 2009

Network Security Protocols and Defensive Mechanisms

John Mitchell

Plan for today

- Network protocol security
  - IPSEC
  - BGP instability and S-BGP
  - DNS rebinding and DNSSEC
  - Wireless security – 802.11i/WPA2
- Standard network perimeter defenses
  - Firewall
    - Packet filter (stateless, stateful), Application layer proxies
  - Traffic shaping
  - Intrusion detection
    - Anomaly and misuse detection

Dan's lecture last Thursday

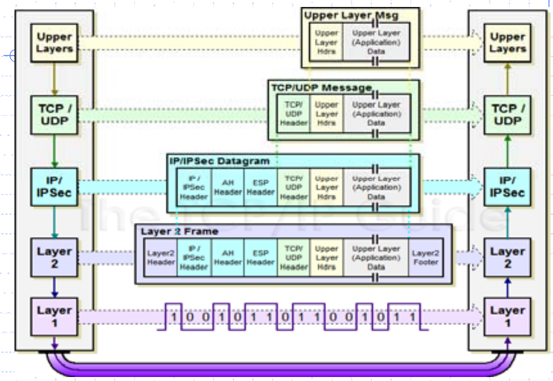
- Basic network protocols
  - IP, TCP, UDP, BGP, DNS
- Problems with them
  - No SRC authentication: can't tell where from
  - Packet sniffing
  - Connection spoofing, sequence numbers
  - BGP: advertise bad routes or close good ones
  - DNS: cache poisoning, rebinding

(out of time; cover today)

IPSEC

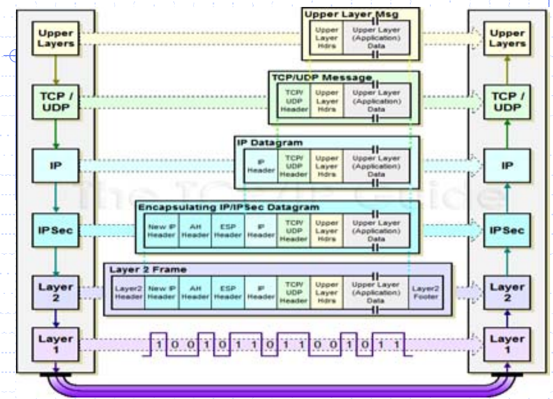
- Security extensions for IPv4 and IPv6
- IP Authentication Header (AH)
  - Authentication and integrity of payload and header
- IP Encapsulating Security Protocol (ESP)
  - Confidentiality of payload
- ESP with optional ICV (integrity check value)
  - Confidentiality, authentication and integrity of payload

IPSec Transport Mode Operation



http://www.tcpipguide.com/free/t_IPSecModesTransportandTunnel.htm

IPSec Tunnel Mode Operation

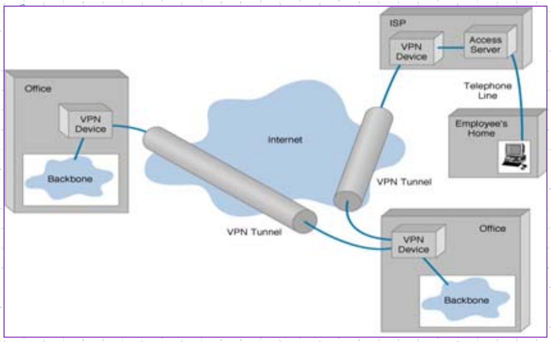


VPN

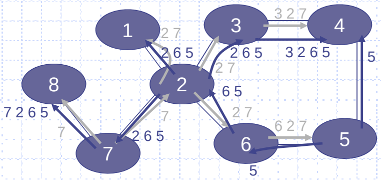
- Three different modes of use:
  - Remote access client connections
  - LAN-to-LAN internetworking
  - Controlled access within an intranet
- Several different protocols
  - PPTP – Point-to-point tunneling protocol
  - L2TP – Layer-2 tunneling protocol
  - IPsec (Layer-3: network layer)

Data layer

Generic diagram



BGP example [D. Wetherall]



- Transit: 2 provides transit for 7
- Algorithm seems to work OK in practice
  - BGP is does not respond well to frequent node outages

BGP Security Issues

- BGP is the critical infrastructure for Internet, the basis for all inter-ISP routing
- Benign configuration errors affect about 1% of all routing table entries at any time
- The current system is highly vulnerable to human errors, and a wide range of malicious attacks
  - links
  - routers
  - management stations
- MD5 MAC is rarely used, perhaps due to lack of automated key management, and it addresses only one class of attacks

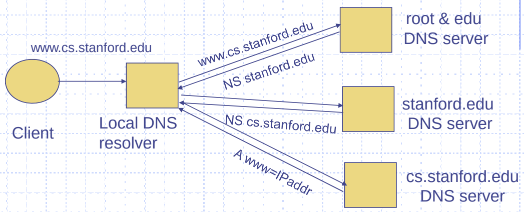
Slide: Steve Kent

S-BGP Design Overview

- **IPsec:** secure point-to-point router communication
- **Public Key Infrastructure:** an authorization framework for all S-BGP entities
- **Attestations:** digitally-signed authorizations to advertise specified address blocks
- Validation of UPDATEs based on a new path attribute, using PKI certificates and attestations
- **Repositories** for distribution of certificates, CRLs, and address attestations
- Tools for ISPs to manage address attestations, process certificates & CRLs, etc.

Slide: Steve Kent

DNS Lookup Example

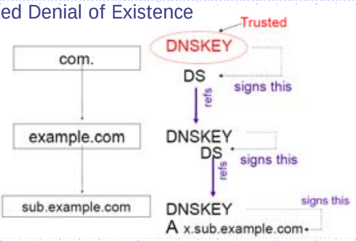


DNS record types (partial list):
- NS: name server (points to other server)
- A: address record (contains IP address)
- MX: address in charge of handling email
- TXT: generic text (e.g. used to distribute site public keys (DKIM))

http://www.nanog.org/mtg-0410/pdf/crocker.pdf

DNSSEC

- Protocol Extensions to DNS provide
  - Data Integrity
  - Origin Authentication of DNS data
  - Authenticated Denial of Existence



Some DNSSEC Issues

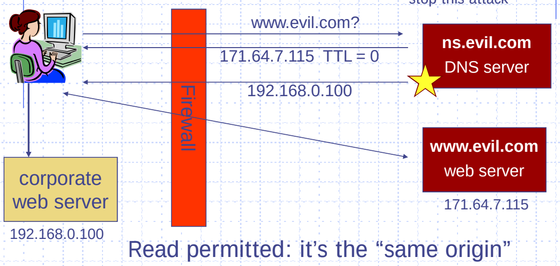
- Root zone key rollover
  - Trust in key is established by DS Resource Record (RR)
    - DS RR of a child zone is stored in its parent zone
    - Carries a "digest" that can uniquely authenticate that DNSKEY
  - Root public key relies on communication "out-of-band" to DNS
  - Lots of politics about who gets to operate DNS root
- What about host names that don't exist in a zone?
  - Simple "does not exist" message would allow replay
  - Better: if name is not in zone, return a "gap-spanning" NSEC RR that gives nearest names before and after the queried name
  - NSEC record lets attacker enumerate a zone
  - Better: NSEC3 record
    - Cryptographically hashes the names, orders the hashes,
    - Uses hashes as in NSEC.

[DWF'96, R'01]

DNS Rebinding Attack

<iframe src="**http://www.evil.com**">

DNS-SEC cannot stop this attack

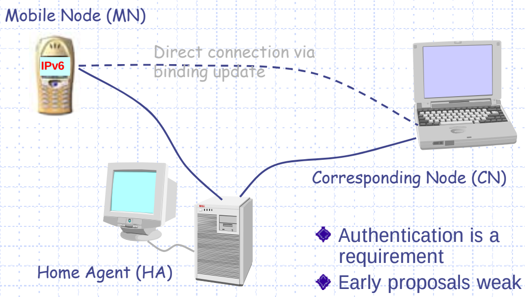


Read permitted: it's the "same origin"

DNS Rebinding Defenses

- Browser mitigation: DNS Pinning
  - Refuse to switch to a new IP
  - Interacts poorly with proxies, VPN, dynamic DNS, …
  - Not consistently implemented in any browser
- Server-side defenses
  - Check Host header for unrecognized domains
  - Authenticate users with something other than IP
- Firewall defenses
  - External names can't resolve to internal addresses
  - Protects browsers inside the organization

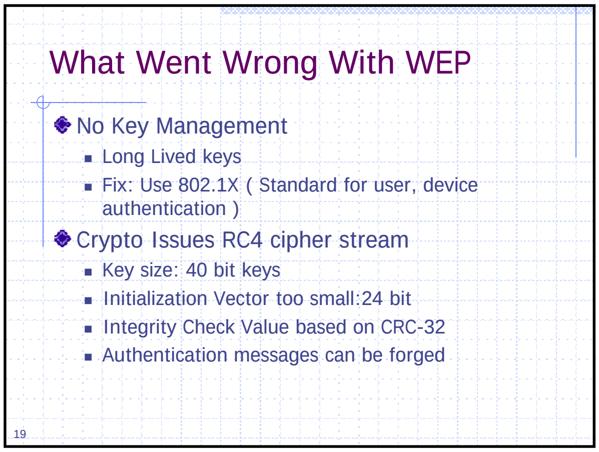
Mobile IPv6 Architecture



- Authentication is a requirement
- Early proposals weak

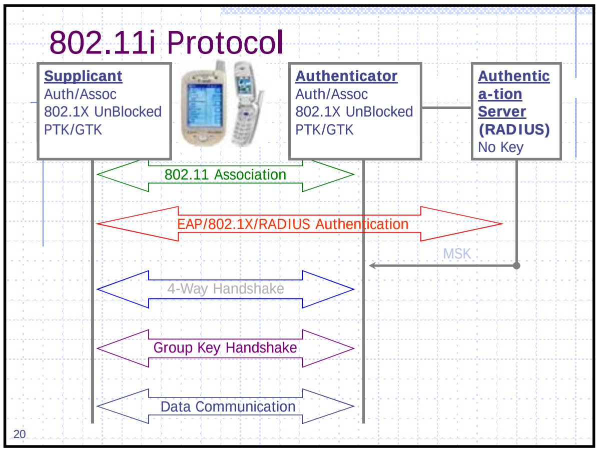
Wireless Access Evolution

- 802.11 (Wired Equivalent Protocol)
  - Authentication: Open system (SSID) and Shared Key
  - Authorization: some vendor use MAC address filtering
  - Confidentiality/Integrity: Completely insecure
- WPA: Wi-Fi Protected Access
  - Authentication: 802.1X
  - Confidentiality/Integrity: TKIP
  - Reuse legacy hardware, still problematic
- IEEE 802.11i (Ratified 2004): WPA2
  - Mutual authentication
  - Data confidentiality and integrity
  - Key management
  - Availability
  - CCMP: AES-based authenticated encryption (integrity,confidentiality)



What Went Wrong With WEP

- No Key Management
  - Long Lived keys
  - Fix: Use 802.1X (Standard for user, device authentication)
- Crypto Issues RC4 cipher stream
  - Key size: 40 bit keys
  - Initialization Vector too small:24 bit
  - Integrity Check Value based on CRC-32
  - Authentication messages can be forged

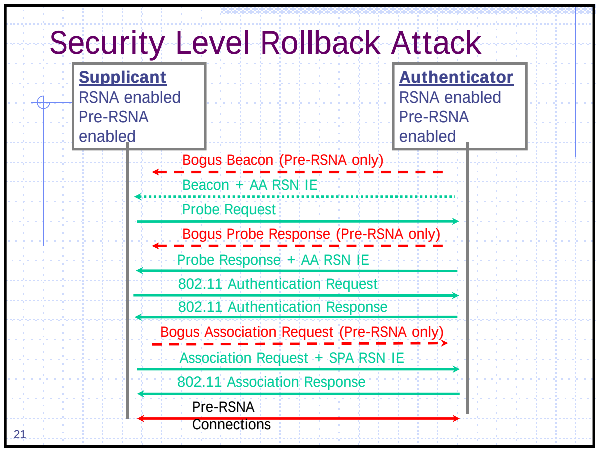


802.11i Protocol

Supplicant
Auth/Assoc
802.1X UnBlocked
PTK/GTK

Authenticator
Auth/Assoc
802.1X UnBlocked
PTK/GTK

Authentica-tion Server (RADIUS)
No Key

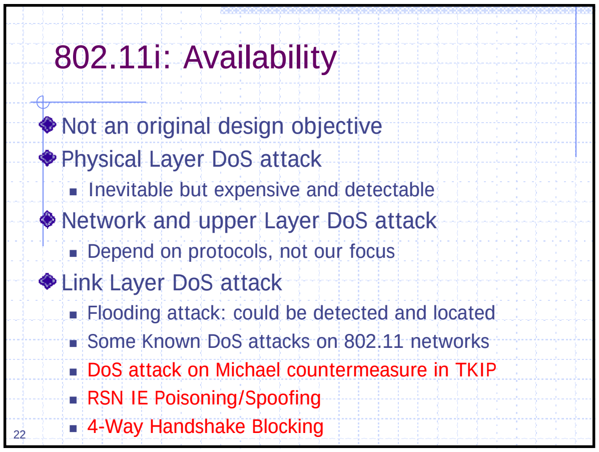


Security Level Rollback Attack

Supplicant
RSNA enabled
Pre-RSNA
enabled

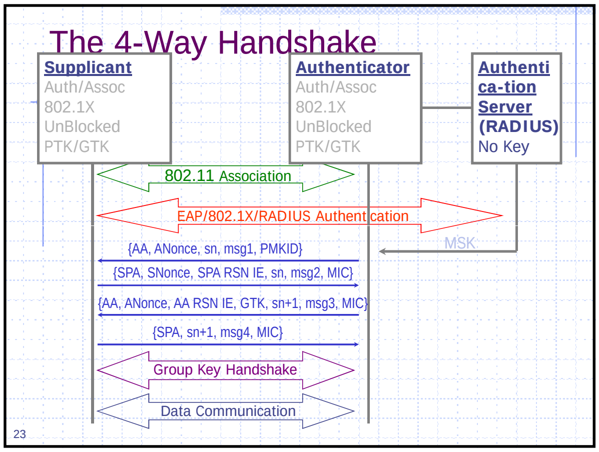
Authenticator
RSNA enabled
Pre-RSNA
enabled

Bogus Beacon (Pre-RSNA only)
Beacon + AA RSN IE
Probe Request
Bogus Probe Response (Pre-RSNA only)
Probe Response + AA RSN IE
802.11 Authentication Request
802.11 Authentication Response
Bogus Association Request (Pre-RSNA only)
Association Request + SPA RSN IE
802.11 Association Response
Pre-RSNA
Connections



802.11i: Availability

- Not an original design objective
- Physical Layer DoS attack
  - Inevitable but expensive and detectable
- Network and upper Layer DoS attack
  - Depend on protocols, not our focus
- Link Layer DoS attack
  - Flooding attack: could be detected and located
  - Some Known DoS attacks on 802.11 networks
  - DoS attack on Michael countermeasure in TKIP
  - RSN IE Poisoning/Spoofing
  - 4-Way Handshake Blocking



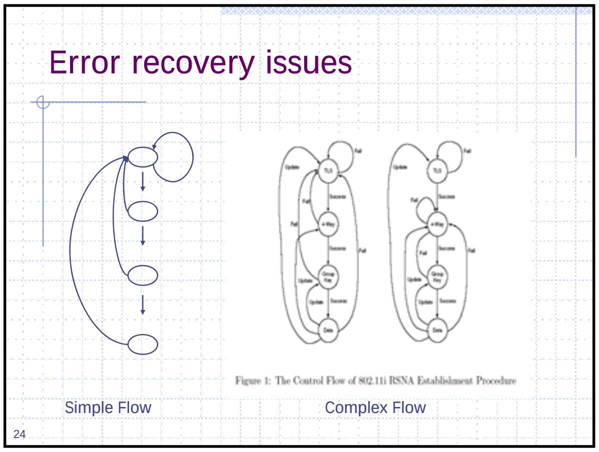
The 4-Way Handshake

Supplicant
Auth/Assoc
802.1X
UnBlocked
PTK/GTK

Authenticator
Auth/Assoc
802.1X
UnBlocked
PTK/GTK

Authentica-tion Server (RADIUS)
No Key

802.11 Association
EAP/802.1X/RADIUS Authentication
MSK
{AA, ANonce, sn, msg1, PMKID}
{SPA, SNonce, SPA RSN IE, sn, msg2, MIC}
{AA, ANonce, AA RSN IE, GTK, sn+1, msg3, MIC}
{SPA, sn+1, msg4, MIC}
Group Key Handshake
Data Communication



Error recovery issues

Figure 1: The Control Flow of 802.11i RSNA Establishment Procedure

Simple Flow

Complex Flow

Summary of 802.11i Design Issues

ATTACKS	SOLUTIONS
security rollback	supplicant *manually* choose security; authenticator restrict pre-RSNA to only insensitive data.
reflection attack	each participant plays the role of either authenticator or supplicant; if both, use different PMKs.
attack on Michael countermeasures	cease connections for a specific time instead of re-key and deauthentication; update TSC before MIC and after FCS, ICV are validated.
RSN IE poisoning	Authenticate Beacon and Probe Response frame; Confirm RSN IE in an earlier stage; Relax the condition of RSN IE confirmation.
4-way handshake blocking	adopt random-drop queue, not so effective; authenticate Message 1, packet format modified; re-use supplicant nonce, eliminate memory DoS.

Announcements

- Project 2 out today
  - Due in two parts over next two weeks
- Discussion section Friday
  - Will cover background for project

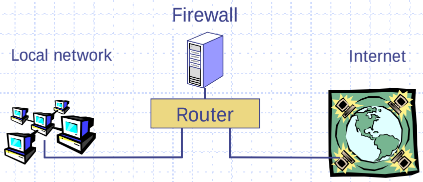
Perimeter and Internal Defenses

- Commonly deployed defenses
  - Perimeter defenses – Firewall, IDS
    - Protect local area network and hosts
    - Keep external threats from internal network
  - Internal defenses – Virus scanning
    - Protect hosts from threats that get through the perimeter defenses
  - Extend the "perimeter" – VPN
- Common practices, but could be improved
  - Internal threats are significant
    - Unhappy employees
    - Compromised hosts

Rest of this lecture

Basic Firewall Concept

- Separate local area net from internet

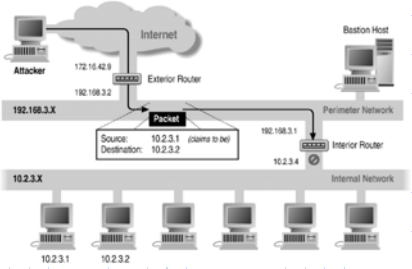


All packets between LAN and internet routed through firewall

Packet Filtering

- Uses transport-layer information only
  - IP Source Address, Destination Address
  - Protocol (TCP, UDP, ICMP, etc)
  - TCP or UDP source & destination ports
  - TCP Flags (SYN, ACK, FIN, RST, PSH, etc)
  - ICMP message type
- Examples
  - DNS uses port 53
    - Block incoming port 53 packets except known trusted servers
- Issues
  - Stateful filtering
  - Encapsulation: address translation, other complications
  - Fragmentation

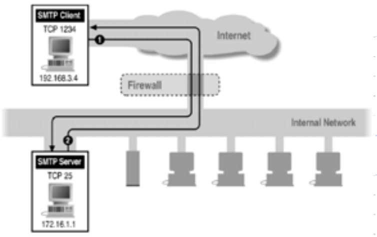
Source/Destination Address Forgery



More about networking: port numbering

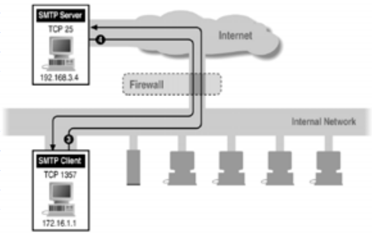
- TCP connection
  - Server port uses number less than 1024
  - Client port uses number between 1024 and 16383
- Permanent assignment
  - Ports <1024 assigned permanently
    - 20,21 for FTP 23 for Telnet
    - 25 for server SMTP 80 for HTTP
- Variable use
  - Ports >1024 must be available for client to make connection
  - Limitation for stateless packet filtering
    - If client wants port 2048, firewall must allow incoming traffic
  - Better: stateful filtering knows outgoing requests
    - Only allow incoming traffic on high port to a machine that has initiated an outgoing request on low port

Filtering Example: Inbound SMTP



Can block external request to internal server based on port number

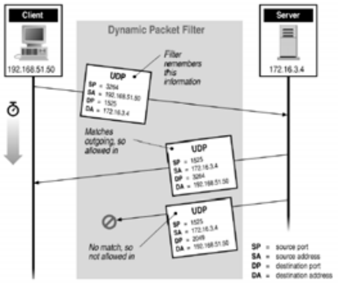
Filtering Example: Outbound SMTP



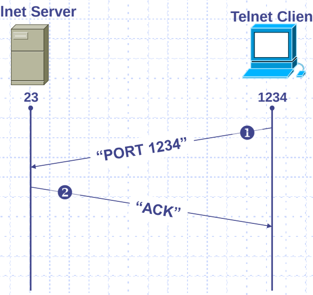
Known low port out, arbitrary high port in

If firewall blocks incoming port 1357 traffic then connection fails

Stateful or Dynamic Packet Filtering



Telnet

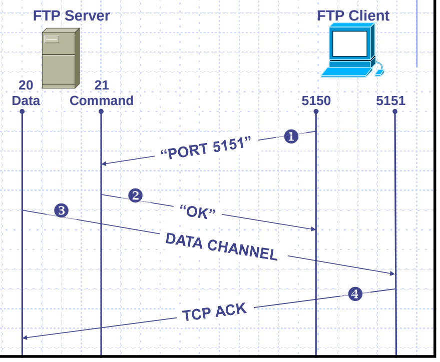


❶ Client opens channel to server; tells server its port number. The ACK bit is not set while establishing the connection but will be set on the remaining packets

❷ Server acknowledges

Stateful filtering can use this pattern to identify legitimate sessions

FTP



❶ Client opens command channel to server; tells server second port number

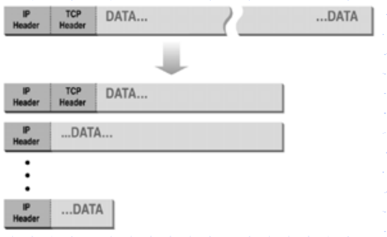
❷ Server acknowledges

❸ Server opens data channel to client's second port

❹ Client acknowledges

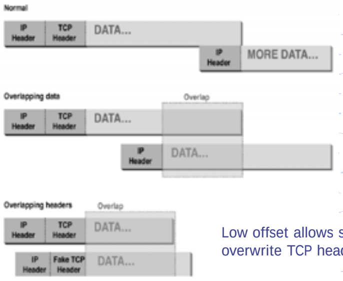
Complication for firewalls

Normal IP Fragmentation



Flags and offset inside IP header indicate packet fragmentation

Abnormal Fragmentation



Low offset allows second packet to overwrite TCP header at receiving host

Packet Fragmentation Attack

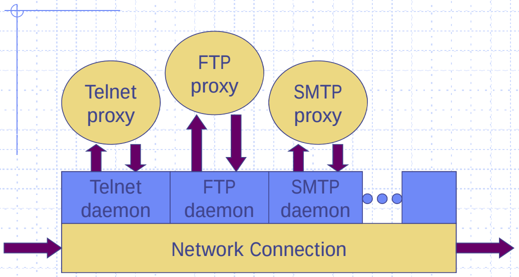
- Firewall configuration
  - TCP port 23 is blocked but SMTP port 25 is allowed
- First packet
  - Fragmentation Offset = 0.
  - DF bit = 0 : "May Fragment"
  - MF bit = 1 : "More Fragments"
  - Destination Port = 25. TCP port 25 is allowed, so firewall allows packet
- Second packet
  - Fragmentation Offset = 1: second packet overwrites all but first 8 bits of the first packet
  - DF bit = 0 : "May Fragment"
  - MF bit = 0 : "Last Fragment."
  - Destination Port = 23. Normally be blocked, but sneaks by!
- What happens
  - Firewall ignores second packet “TCP header” because it is fragment of first
  - At host, packet reassembled and received at port 23

Beyond packet filtering

Proxying Firewall

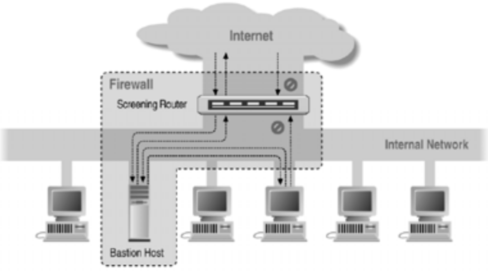
- Application-level proxies
  - Tailored to http, ftp, smtp, etc.
  - Some protocols easier to proxy than others
- Policy embedded in proxy programs
  - Proxies filter incoming, outgoing packets
  - Reconstruct application-layer messages
  - Can filter specific application-layer commands, etc.
    - Example: only allow specific ftp commands
    - Other examples: ?
- Several network locations – see next slides

Firewall with application proxies

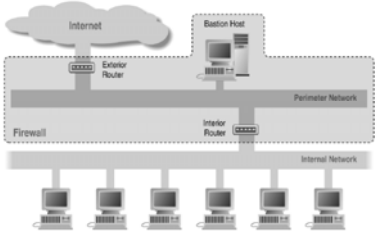


Daemon spawns proxy when communication detected …

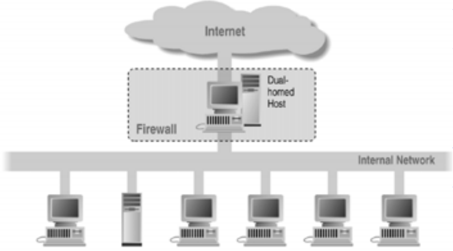
Screened Host Architecture



Screened Subnet Using Two Routers



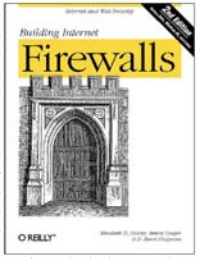
Dual Homed Host Architecture



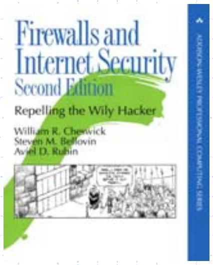
Application-level proxies

- Enforce policy for specific protocols
  - E.g., Virus scanning for SMTP
    - Need to understand MIME, encoding, Zip archives
  - Flexible approach, but may introduce network delays
- “Batch” protocols are natural to proxy
  - SMTP (E-Mail) NNTP (Net news)
  - DNS (Domain Name System) NTP (Network Time Protocol
- Must protect host running protocol stack
  - Disable all non-required services; keep it simple
  - Install/modify services you want
  - Run security audit to establish baseline
  - Be prepared for the system to be compromised

References



Elizabeth D. Zwicky
Simon Cooper
D. Brent Chapman



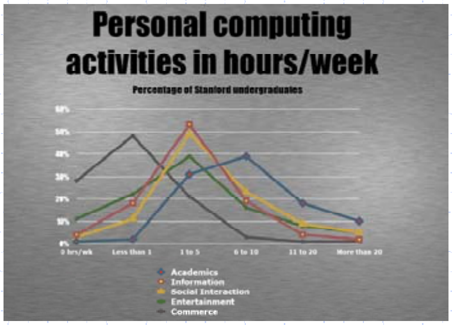
William R Cheswick
Steven M Bellovin
Aviel D Rubin

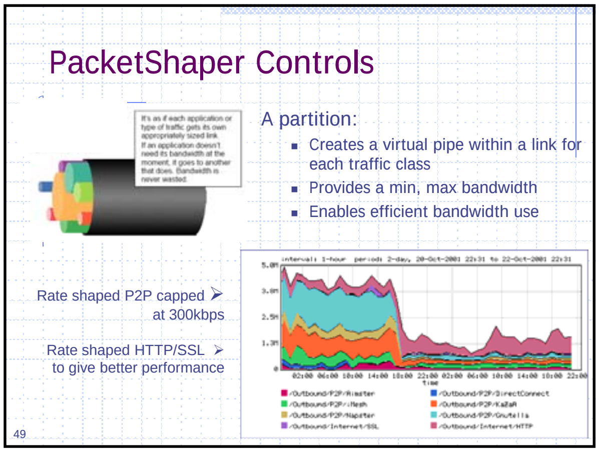
Traffic Shaping

- Traditional firewall
  - Allow traffic or not
- Traffic shaping
  - Limit certain kinds of traffic
  - Can differentiate by host addr, protocol, etc
  - Multi-Protocol Label Switching (MPLS)
    - Label traffic flows at the edge of the network and let core routers identify the required class of service

- The real issue here on Campus:
  - P2P file sharing takes a lot of bandwidth
  - 1/3 of network bandwidth consumed by BitTorrent
    - Students: what are BitTorrent, Gnutella, Kazaa, … used for?

Stanford computer use





PacketShaper Controls

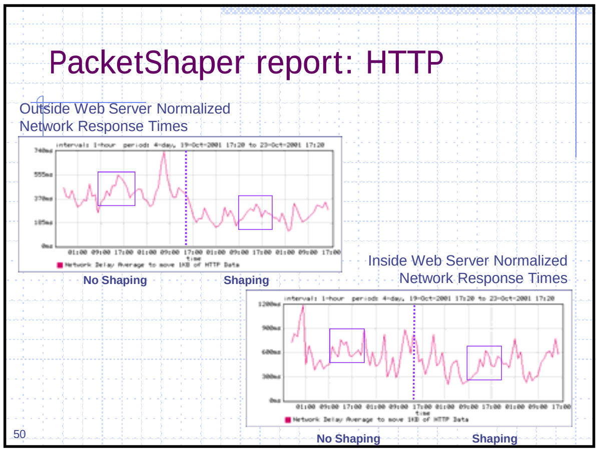
It's as if each application or type of traffic gets its own appropriately sized link. If an application doesn't need its bandwidth at the moment, it goes to another that does. Bandwidth is never wasted.

A partition:

- Creates a virtual pipe within a link for each traffic class
- Provides a min, max bandwidth
- Enables efficient bandwidth use

Rate shaped P2P capped ➢ at 300kbps

Rate shaped HTTP/SSL ➢ to give better performance



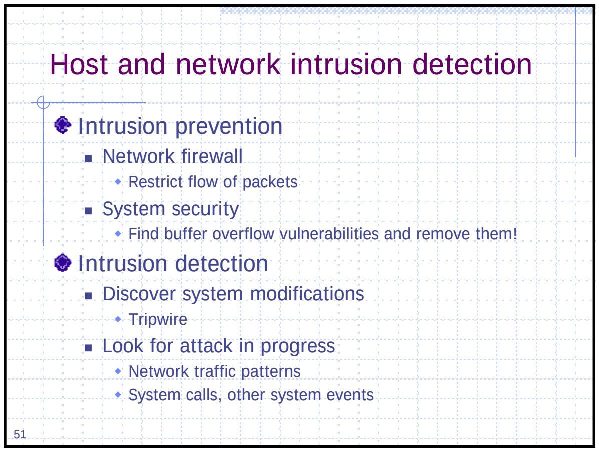
PacketShaper report: HTTP

Outside Web Server Normalized Network Response Times

No Shaping Shaping

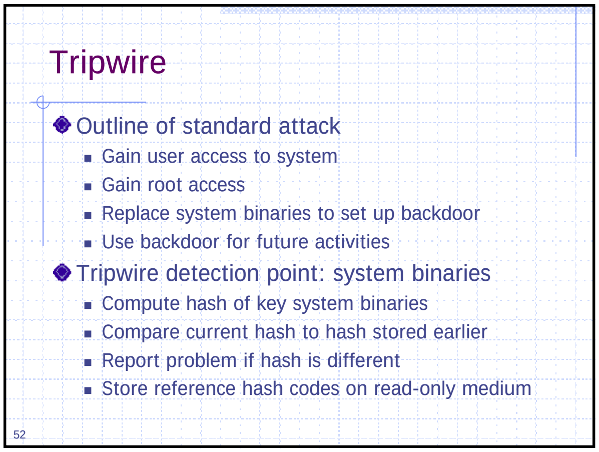
Inside Web Server Normalized Network Response Times

No Shaping Shaping



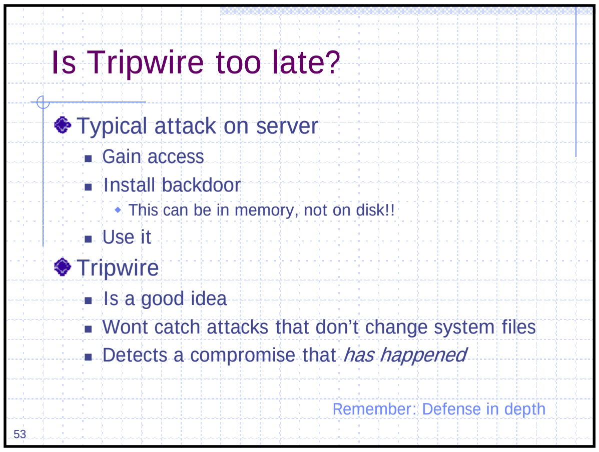
Host and network intrusion detection

- Intrusion prevention
  - Network firewall
    - Restrict flow of packets
  - System security
    - Find buffer overflow vulnerabilities and remove them!
- Intrusion detection
  - Discover system modifications
    - Tripwire
  - Look for attack in progress
    - Network traffic patterns
    - System calls, other system events



Tripwire

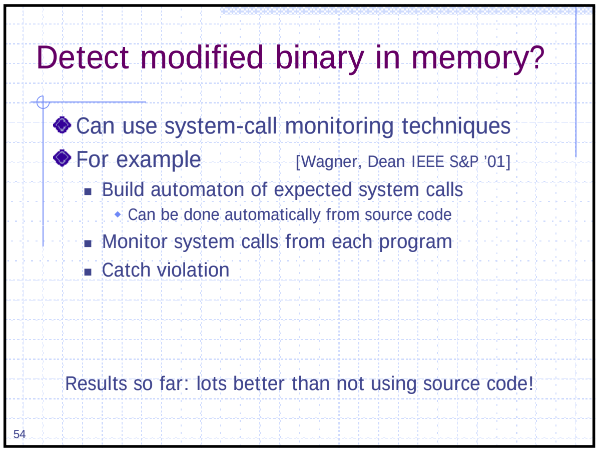
- Outline of standard attack
  - Gain user access to system
  - Gain root access
  - Replace system binaries to set up backdoor
  - Use backdoor for future activities
- Tripwire detection point: system binaries
  - Compute hash of key system binaries
  - Compare current hash to hash stored earlier
  - Report problem if hash is different
  - Store reference hash codes on read-only medium



Is Tripwire too late?

- Typical attack on server
  - Gain access
  - Install backdoor
    - This can be in memory, not on disk!!
  - Use it
- Tripwire
  - Is a good idea
  - Wont catch attacks that don't change system files
  - Detects a compromise that *has happened*

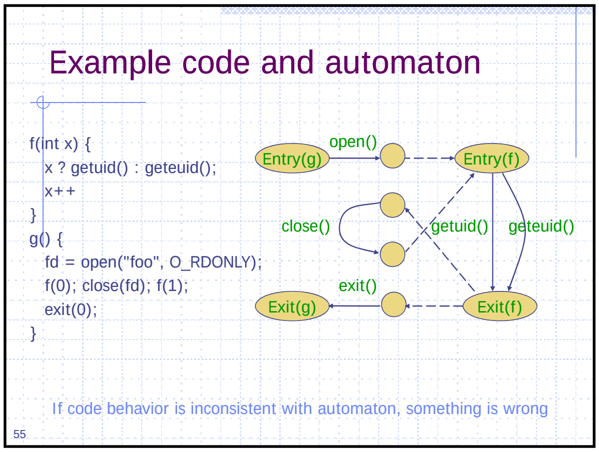
Remember: Defense in depth



Detect modified binary in memory?

- Can use system-call monitoring techniques
- For example [Wagner, Dean IEEE S&P '01]
  - Build automaton of expected system calls
    - Can be done automatically from source code
  - Monitor system calls from each program
  - Catch violation

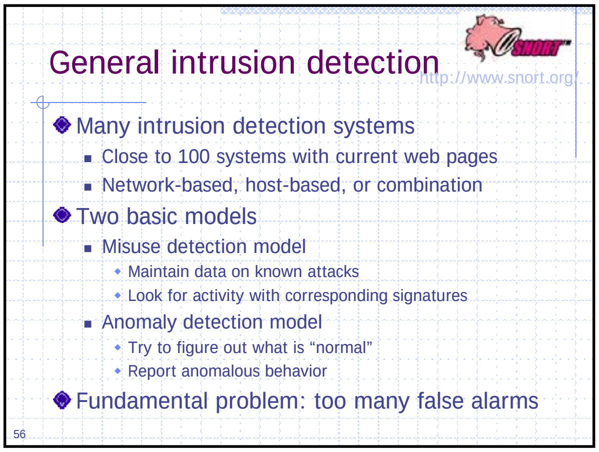
Results so far: lots better than not using source code!



Example code and automaton

```
f(int x) {
   x ? getuid() : geteuid();
   x++
}
g() {
   fd = open("foo", O_RDONLY);
   f(0); close(fd); f(1);
   exit(0);
}
```

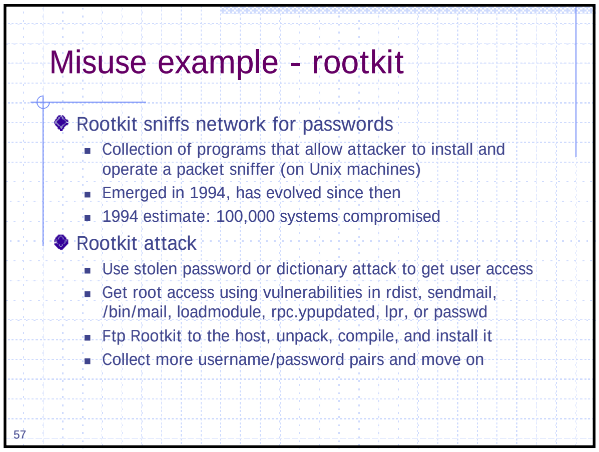
If code behavior is inconsistent with automaton, something is wrong



General intrusion detection

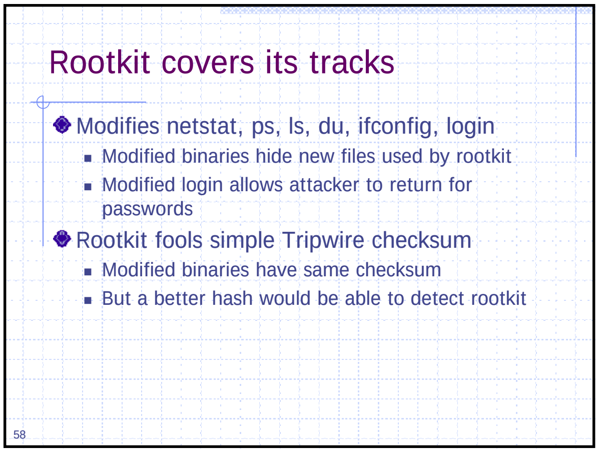
http://www.snort.org/

- Many intrusion detection systems
  - Close to 100 systems with current web pages
  - Network-based, host-based, or combination
- Two basic models
  - Misuse detection model
    - Maintain data on known attacks
    - Look for activity with corresponding signatures
  - Anomaly detection model
    - Try to figure out what is “normal”
    - Report anomalous behavior
- Fundamental problem: too many false alarms



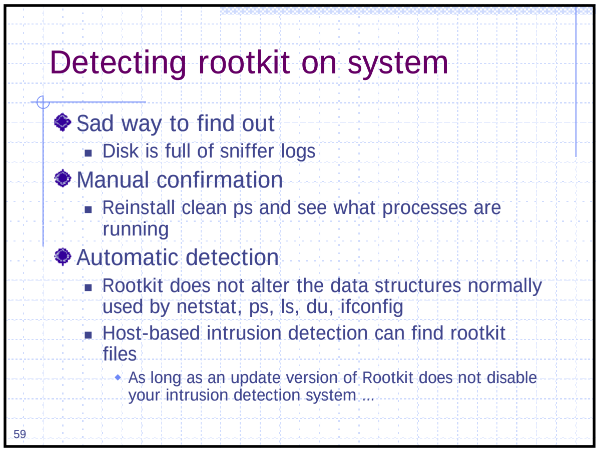
Misuse example - rootkit

- Rootkit sniffs network for passwords
  - Collection of programs that allow attacker to install and operate a packet sniffer (on Unix machines)
  - Emerged in 1994, has evolved since then
  - 1994 estimate: 100,000 systems compromised
- Rootkit attack
  - Use stolen password or dictionary attack to get user access
  - Get root access using vulnerabilities in rdist, sendmail, /bin/mail, loadmodule, rpc.ypupdated, lpr, or passwd
  - Ftp Rootkit to the host, unpack, compile, and install it
  - Collect more username/password pairs and move on



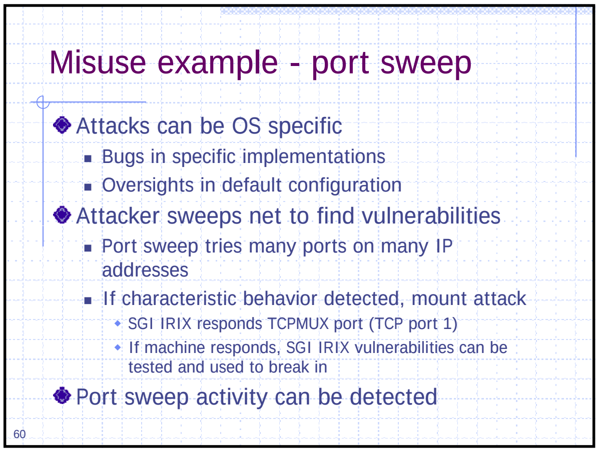
Rootkit covers its tracks

- Modifies netstat, ps, ls, du, ifconfig, login
  - Modified binaries hide new files used by rootkit
  - Modified login allows attacker to return for passwords
- Rootkit fools simple Tripwire checksum
  - Modified binaries have same checksum
  - But a better hash would be able to detect rootkit



Detecting rootkit on system

- Sad way to find out
  - Disk is full of sniffer logs
- Manual confirmation
  - Reinstall clean ps and see what processes are running
- Automatic detection
  - Rootkit does not alter the data structures normally used by netstat, ps, ls, du, ifconfig
  - Host-based intrusion detection can find rootkit files
    - As long as an update version of Rootkit does not disable your intrusion detection system ...



Misuse example - port sweep

- Attacks can be OS specific
  - Bugs in specific implementations
  - Oversights in default configuration
- Attacker sweeps net to find vulnerabilities
  - Port sweep tries many ports on many IP addresses
  - If characteristic behavior detected, mount attack
    - SGI IRIX responds TCPMUX port (TCP port 1)
    - If machine responds, SGI IRIX vulnerabilities can be tested and used to break in
- Port sweep activity can be detected

Anomaly Detection

- Basic idea
  - Monitor network traffic, system calls
  - Compute statistical properties
  - Report errors if statistics outside established range
- Example – IDES (Denning, SRI)
  - For each user, store daily count of certain activities
    - E.g., Fraction of hours spent reading email
  - Maintain list of counts for several days
  - Report anomaly if count is outside weighted norm

Big problem: most unpredictable user is the most important

[Hofmeyr, Somayaji, Forrest]

Anomaly – sys call sequences

- Build traces during normal run of program
  - Example program behavior (sys calls)

    open read write open mmap write fchmod close
  - Sample traces stored in file (4-call sequences)

    open read write open
    read write open mmap
    write open mmap write
    open mmap write fchmod
    mmap write fchmod close
  - Report anomaly if following sequence observed

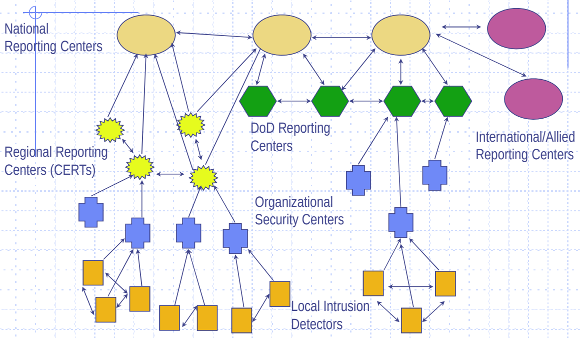
    open read read open mmap write fchmod close

Compute # of mismatches to get mismatch rate

Difficulties in intrusion detection

- Lack of training data
  - Lots of "normal" network, system call data
  - Little data containing realistic attacks, anomalies
- Data drift
  - Statistical methods detect changes in behavior
  - Attacker can attack gradually and incrementally
- Main characteristics not well understood
  - By many measures, attack may be within bounds of "normal" range of activities
- False identifications are very costly
  - Sys Admin spend many hours examining evidence

Strategic Intrusion Assessment [Lunt]



www.blackhat.com/presentations/bh-usa-99/teresa-lunt/tutorial.ppt

Strategic Intrusion Assessment [Lunt]

- Test over two-week period
  - AFIWC's intrusion detectors at 100 AFBs alarmed on 2 million sessions
  - Manual review identified 12,000 suspicious events
  - Further manual review => four actual incidents
- Conclusion
  - Most alarms are false positives
  - Most true positives are trivial incidents
  - Of the significant incidents, most are isolated attacks to be dealt with locally

Summary

- Network protocol security
  - IPSEC
  - BGP instability and S-BGP
  - DNSSEC, DNS rebinding
  - Wireless security – 802.11i/WPA2
- Standard network perimeter defenses
  - Firewall
    - Packet filter (stateless, stateful), Application layer proxies
  - Traffic shaping
  - Intrusion detection
    - Anomaly and misuse detection