Lecture 29: Networking Defenses

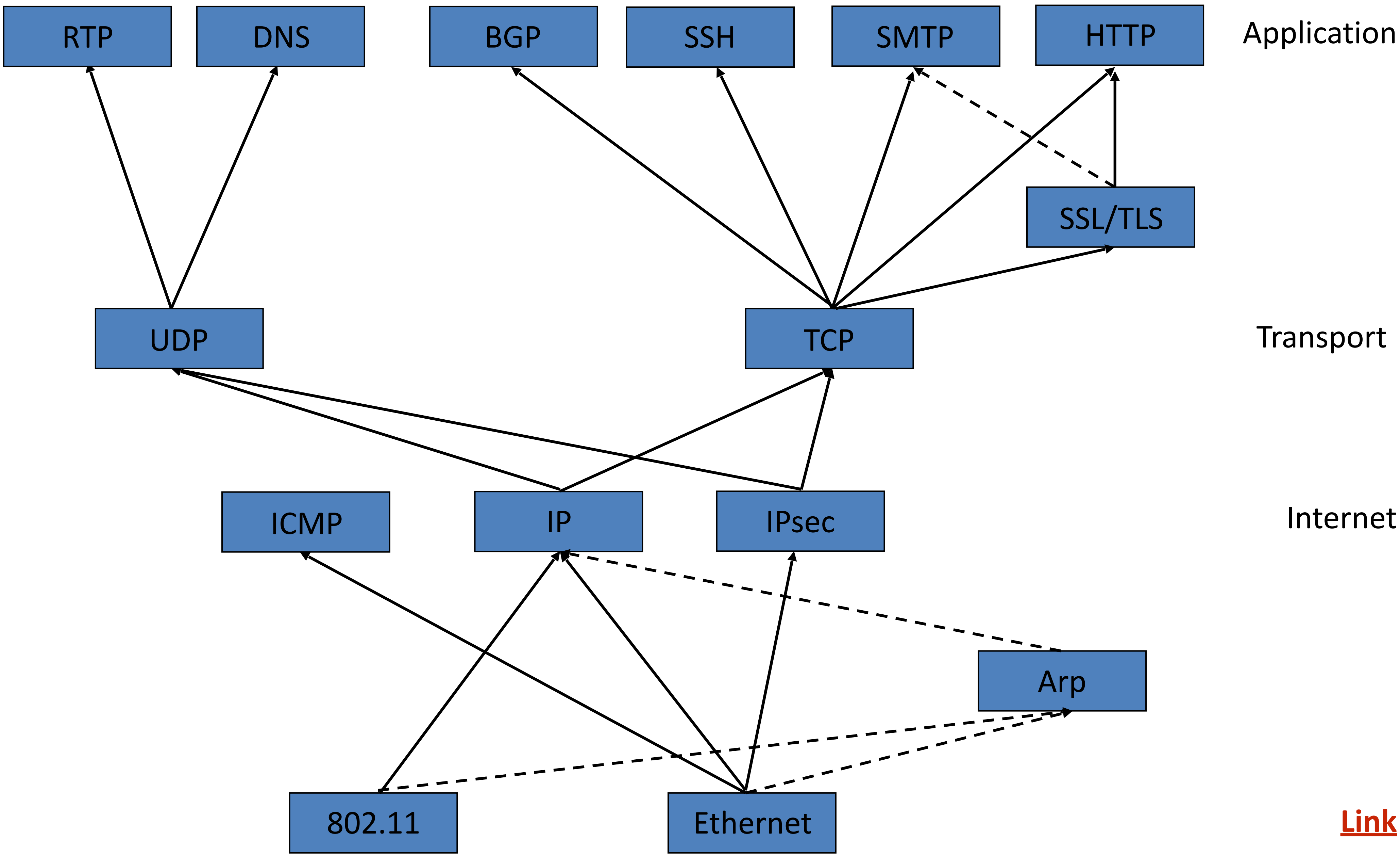
Professor Adam Bates
CS 461 / ECE 422
Fall 2019

Goals for Today

- Learning Objectives:
  - Explore additional approaches to securing networks
- Announcements, etc:
  - Vote on favorite u-pick-em lecture topic (Piazza)
  - Networking CP2: Due Nov 11
  - Final Exam *is* on Dec 13 at 7pm
  - November 5th at 10am in 2405 Siebel Center:
    - "Securing Software-defined Networking Infrastructure" w/ Prof. Guofei Gu (TAMU)

Reminder: Please put away devices at the start of class

Network Stack



IEEE 802.11 security

- War-driving: drive around Bay area, see what 802.11 networks available?
  - More than 9000 accessible from public roadways
  - 85% use no encryption/authentication
  - packet-sniffing and various attacks easy!
- Securing 802.11
  - encryption, authentication
  - first attempt at 802.11 security: Wired Equivalent Privacy (WEP): a failure
  - current attempt: 802.11i

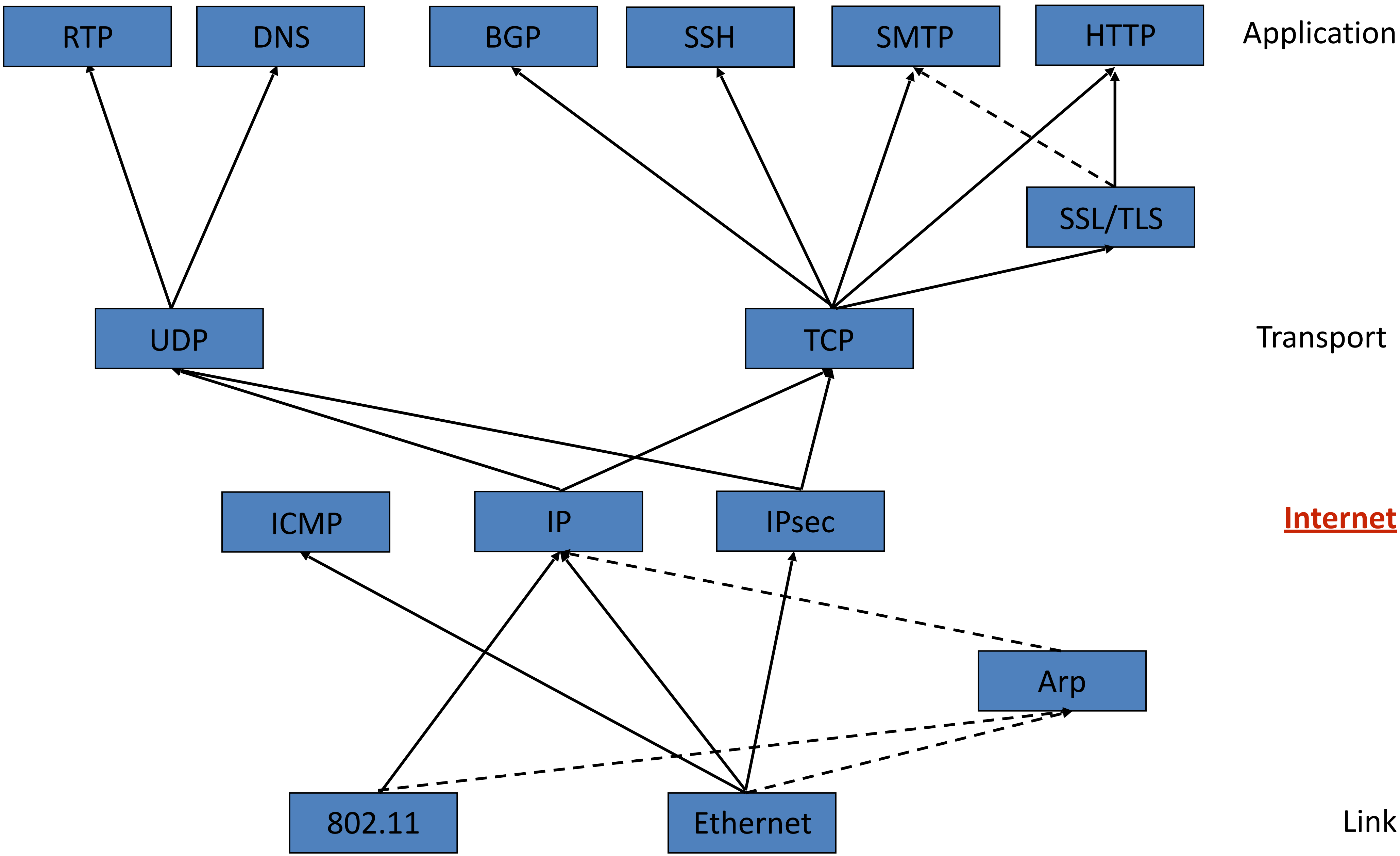
Wired Equivalent Privacy (WEP)

- authentication as in protocol ap4.0
  - host requests authentication from access point
  - access point sends 128 bit nonce
  - host encrypts nonce using shared symmetric key
  - access point decrypts nonce, authenticates host
- no key distribution mechanism
- authentication: knowing the shared key is enough

Breaking 802.11 WEP encryption

- Security hole:
- 24-bit IV, one IV per frame, -> IV's eventually reused
- IV transmitted in plaintext -> IV reuse detected
- Attack:
  - Trudy causes Alice to encrypt known plaintext d_1 d_2 d_3 …
  - Trudy sees: c_i = d_i XOR k_i{IV}
  - Trudy knows ci di, so can compute k_i{IV}
  - Trudy knows encrypting key sequence k_1{IV} k_2{IV} k_3{IV} …
  - Next time IV is used, Trudy can decrypt!

Network Stack



TLS/SSL

- transport layer security to any TCP-based app using SSL services.
- used between Web browsers, servers for e-commerce (shttp).
- security services:
  - server authentication
  - data encryption
  - client authentication (optional)

- server authentication:
  - SSL-enabled browser includes public keys for trusted CAs.
  - Browser requests server certificate, issued by trusted CA.
  - Browser uses CA's public key to extract server's public key from certificate.
- check your browser's security menu to see its trusted CAs.

TLS/SSL

Encrypted SSL session:

- Browser generates *symmetric session key*, encrypts it with server's public key, sends encrypted key to server.
- Using private key, server decrypts session key.
- Browser, server know session key
  - All data sent into TCP socket (by client or server) encrypted with session key.

- SSL: basis of IETF Transport Layer Security (TLS).
- SSL can be used for non-Web applications, e.g., IMAP.
- Client authentication can be done with client certificates.

IPsec: Network Layer Security

- Network-layer secrecy:
  - sending host encrypts the data in IP datagram
  - TCP and UDP segments; ICMP and SNMP messages.
- Network-layer authentication
  - destination host can authenticate source IP address
- Two principle protocols:
  - authentication header (AH) protocol
  - encapsulation security payload (ESP) protocol
- For both AH and ESP, source, destination handshake:
  - create network-layer logical channel called a security association (SA)
- Each SA unidirectional.
- Uniquely determined by:
  - security protocol (AH or ESP)
  - source IP address
  - 32-bit connection ID

Why is IPSec necessary?

Authentication Headers

- provides source authentication, data integrity, no confidentiality
- AH header inserted between IP header, data field.
- protocol field: 51
- intermediate routers process datagrams as usual

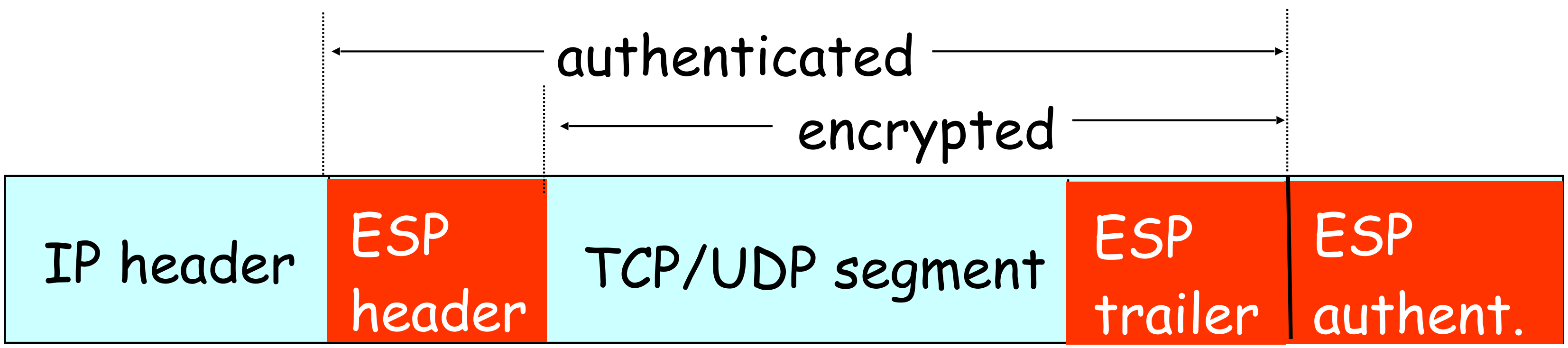
AH header includes:

- connection identifier
- authentication data: source-signed message digest calculated over original IP datagram.
- next header field: specifies type of data (e.g., TCP, UDP, ICMP)

IP header	AH header	data (e.g., TCP, UDP segment)

Encapsulating Security Payload (ESP)

- Can provide source authentication, data integrity, and confidentiality through encryption protection for IP packets
- ESP authentication field is similar to AH authentication field.
- Intermediate routers process datagrams as usual
- Protocol Field: 50.



What is a VPN?

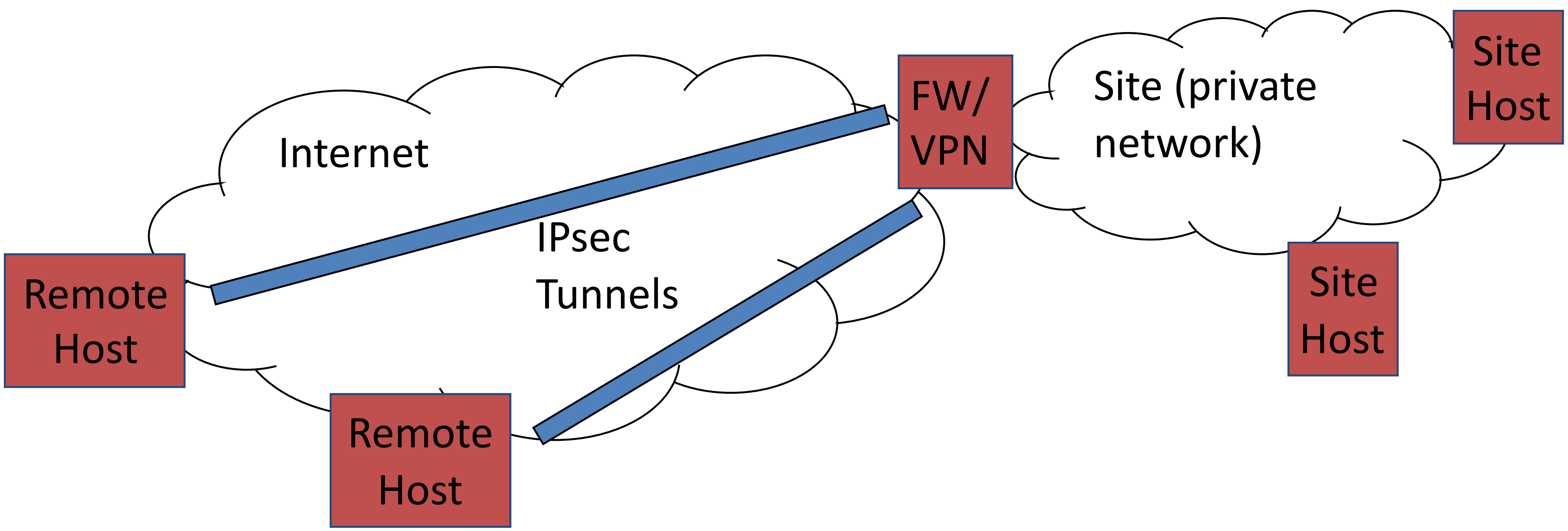
- Making a shared network look like a private network
- Why do this?
  - Private networks have all kinds of advantages
  - But building a private network is expensive
    - (cheaper to have shared resources rather than dedicated)

IP VPN benefits

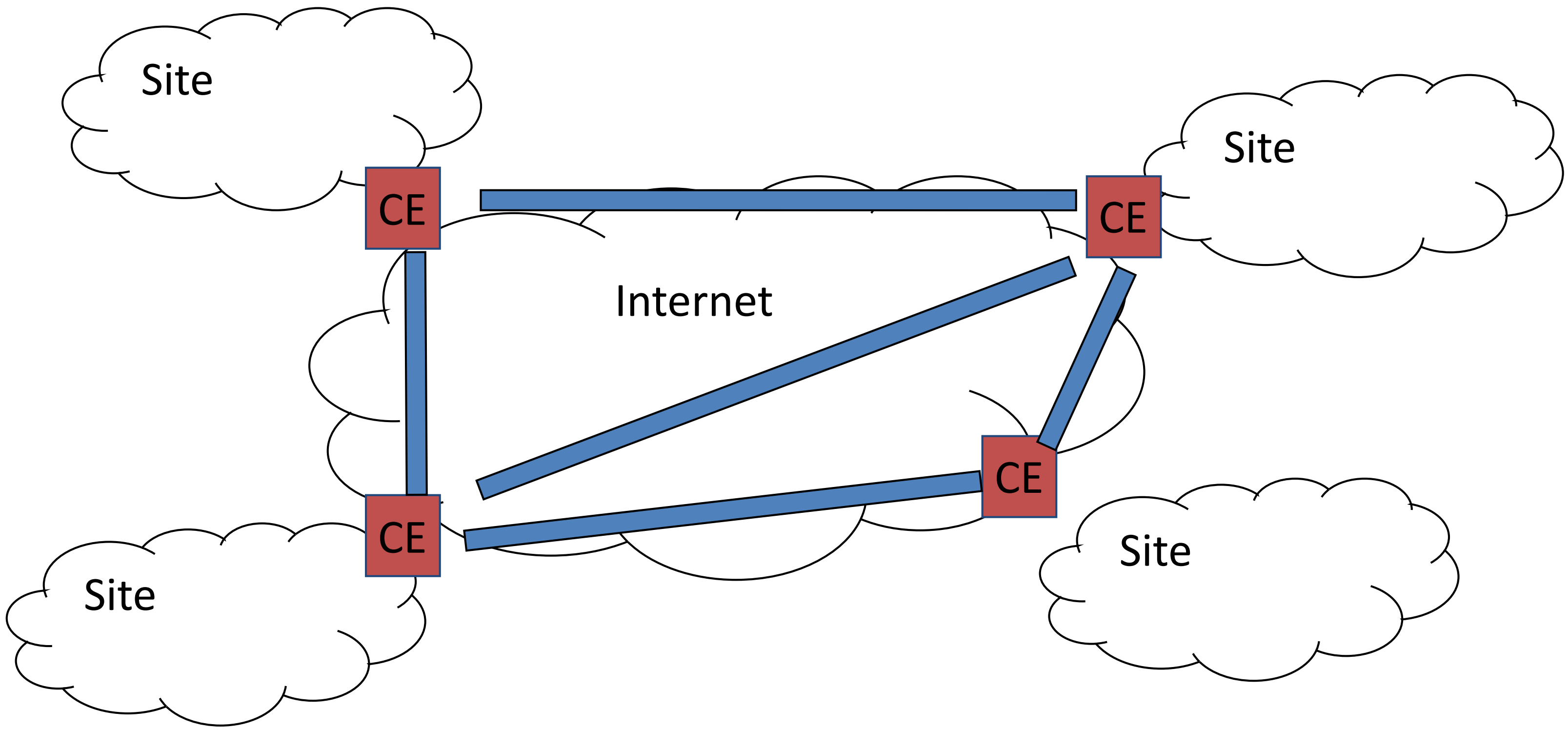
- IP not really global (private addresses)
  - VPN makes separated IP sites look like one private IP network
- Security
- Bandwidth guarantees across ISP
  - QoS, SLAs
- Simplified network operation
  - ISP can do the routing for you

End-to-end VPNs

- Solves problem of how to connect remote hosts to a firewalled network



Customer-based Network VPNs

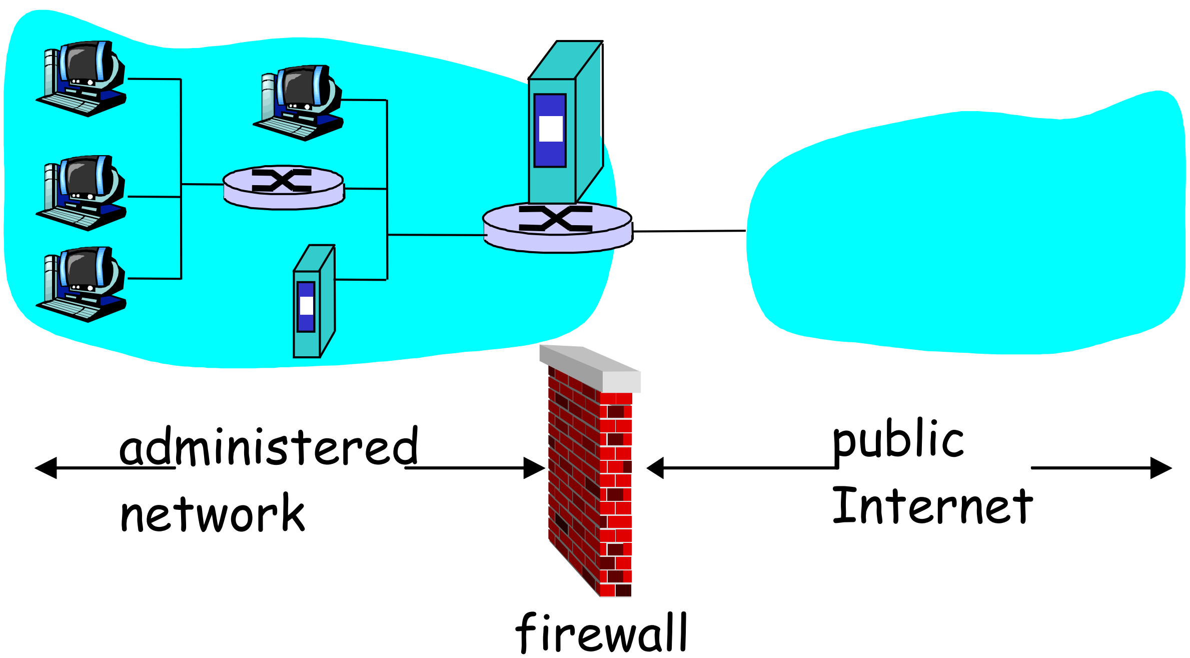


Customer buys own equipment, configures IPsec tunnels over the global internet, manages addressing and routing. ISP plays no role.

Firewalls

firewall

isolates organization's internal net from larger Internet, allowing some packets to pass, blocking others.

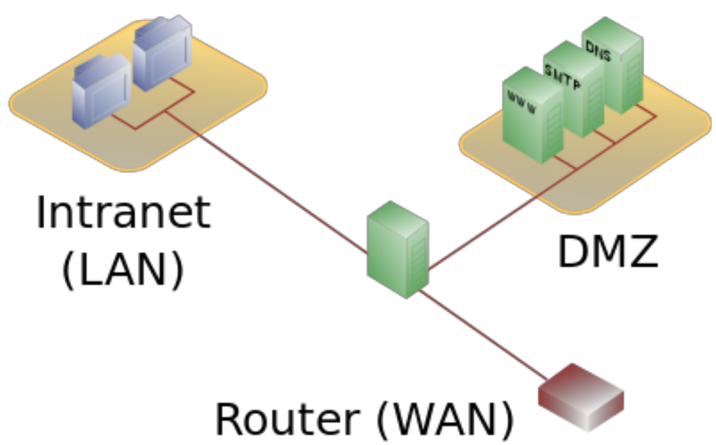


Firewalls: Why

- *Minimize organizational attack surface!*
- prevent denial of service attacks:
  - SYN flooding: attacker establishes many bogus TCP connections, no resources left for “real” connections.
- prevent illegal modification/access of internal data.
  - e.g., attacker replaces CIA’s homepage with something else
- allow only authorized access to inside network (set of authenticated users/hosts)
- two types of firewalls:
  - application-level
  - packet-filtering

Firewalls: Why

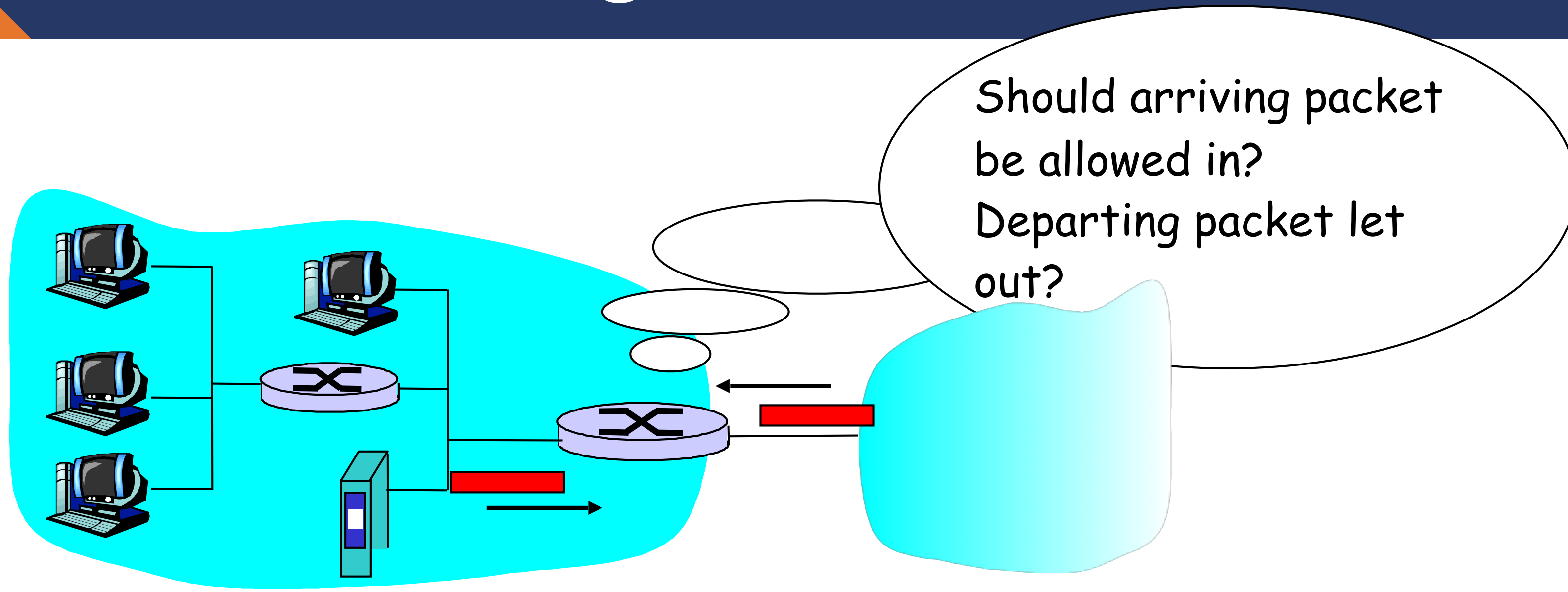
- *Minimize organizational attack surface!*
- Establish a ***Demilitarized Zone (DMZ)***: A perimeter network that segments external-facing services (e.g., web, email) from the rest of the organizational network.
- With DMZ's, if web server is compromised, threat to broader organization is mitigated.



Firewalls: not just protection from outside attackers

- Bandwidth control
  - Block high bandwidth applications
  - Pointcast, Napster
- Employee network usage control
  - Block games, pornography, non-business uses
- Privacy
  - Don't let outside see what you have, how big you are, etc.
  - Similar to making corporate phone directory proprietary

Packet Filtering



- internal network connected to Internet via router firewall
- router filters packet-by-packet, decision to forward/drop packet based on:
  - source IP address, destination IP address
  - TCP/UDP source and destination port numbers
  - ICMP message type
  - TCP SYN and ACK bits

Simple firewall policy configuration

Source	Dest	App	Action
any-inside	dmz-mail	SMTP	allow
any-inside	any-outside	SMTP	drop
any-inside	any-outside	HTTP	allow
any-inside	any-outside	FTP	allow
any-inside	any-outside	any	drop
any-outside	any-inside	any	drop

Firewall Challenge (Ex)

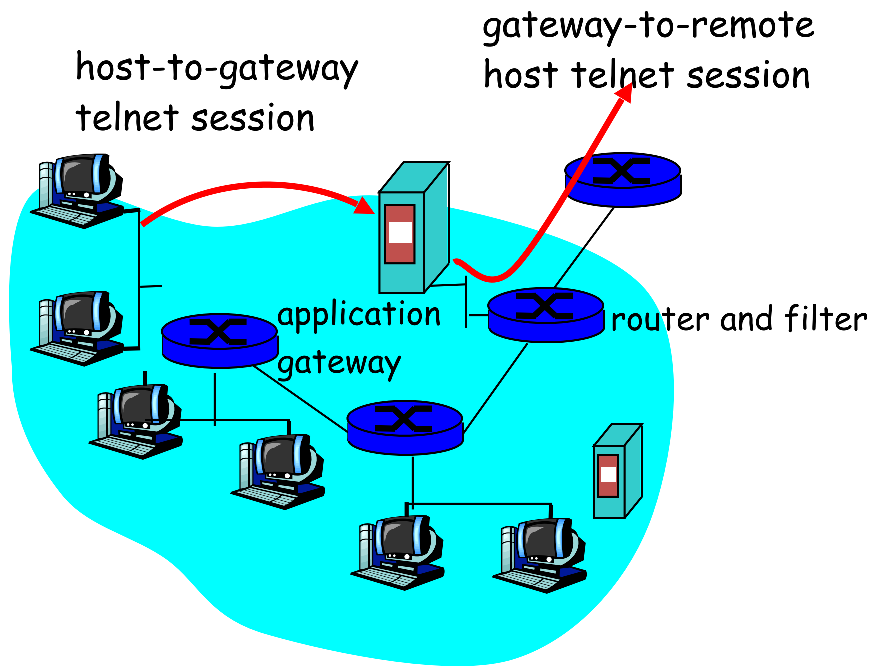
- FTP consists of two flows, control flow and data flow
- Firewall must be smart enough to read control flow, identify subsequent data flow
- True for SIP and other protocols as well

Stateful and stateless firewalls

- Original firewalls were stateless
  - Maintain static filter list, but no per flow state
  - For TCP, only look at SYN
    - Means that non-SYN TCP packets are allowed even if should be blocked
  - No concept of conversation
- Modern firewalls are typically stateful
  - Maintains dynamic list of all allowed flows
  - Better capability, harder to scale

Application gateways

- Filters packets on application data as well as on IP/TCP/UDP fields.
- Example: allow select internal users to telnet outside.



1. Require all telnet users to telnet through gateway.
2. For authorized users, gateway sets up telnet connection to dest host. Gateway relays data between 2 connections
3. Router filter blocks all telnet connections not originating from gateway.

Limitations of firewalls and gateways

- IP spoofing: router can't know if data "really" comes from claimed source
- if multiple app's. need special treatment, each has own app. gateway.
- client software must know how to contact gateway.
  - e.g., must set IP address of proxy in Web browser
- filters often use all or nothing policy for UDP.
- tradeoff: degree of communication with outside world, level of security
- many highly protected sites still suffer from attacks.

Network Intrusion Detection Systems (NIDS)

- "Building burglar alarms for the net"
- Idea: make systems sensitive to threatening actions, and make them capable of alerting authorities when they notice anomalies
- Necessarily post-hoc
- Broad types
  - Statistical analyzers (anomaly based)
  - Rules-based systems, Attack-signature detectors (misuse)
  - Others

Know Your Attacker

- Most attackers run scripts to probe for vulnerabilities, then return later to exploit them
- Probes tend to come in waves as new holes are discovered
- Probes look very different than typical network use
- Actual attack may come long after probe

Paradigms in Intrusion Detection

- **Misuse Detection Intrusion Detection Systems (MD)**
  - define *"what is abnormal"* using attack signatures
  - traffic that matches an attack signature as attack traffic
- **Anomaly Detection Intrusion Detection Systems (AD)**
  - define *"what is normal"* using profiles
  - traffic that does not match the profile as abnormal

The world's simplest NIDS

```
v=listen(frequently-exploited-unused-port);
while(1) {
    s=accept(v, who, howbig);
    notify_the_authorities(s, who, howbig);
    close(s);
}
```

- This won't catch stealth scanners
- Doesn't have a global view
- Can't detect attacks on systems in use
- Surprisingly effective at catching scans nonetheless

Statistical analysis

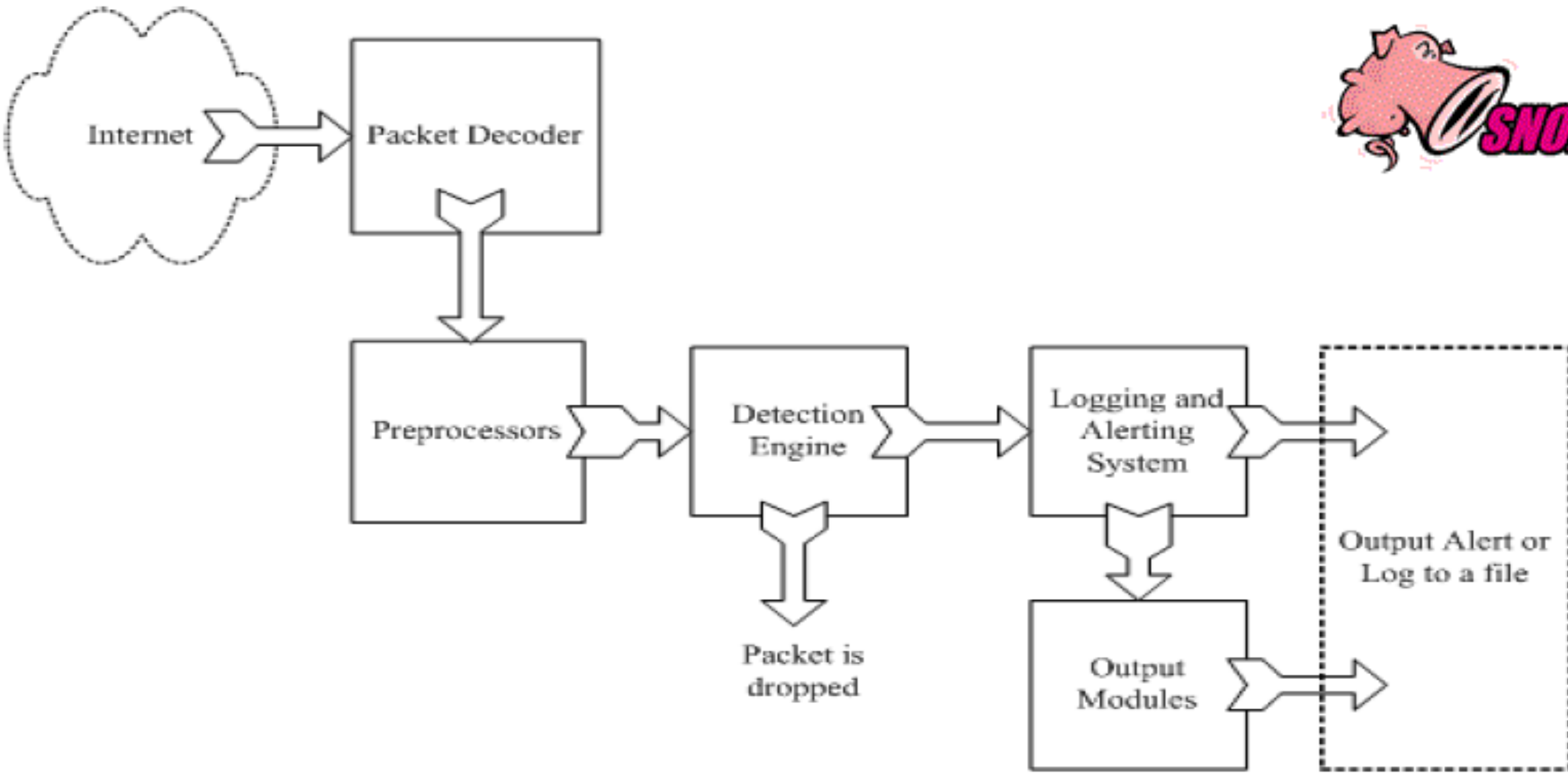
- Constantly capture packets, watch logs, note typical flows
  – I.E. “95% of traffic flows from inside the firewall to outside web services”
  – Set off alarm bells when traffic not matching typical flows is seen
  – Can be a first alert against configuration problems
- Gains a global picture of the system

Rule-based systems

- Monitor logs and network for behavior violating or matching static rules
- Require some knowledge of attack behaviors
- Less prone to false alarms
- Often combined with anomaly detectors

Example: Snort

http://www.snort.org/

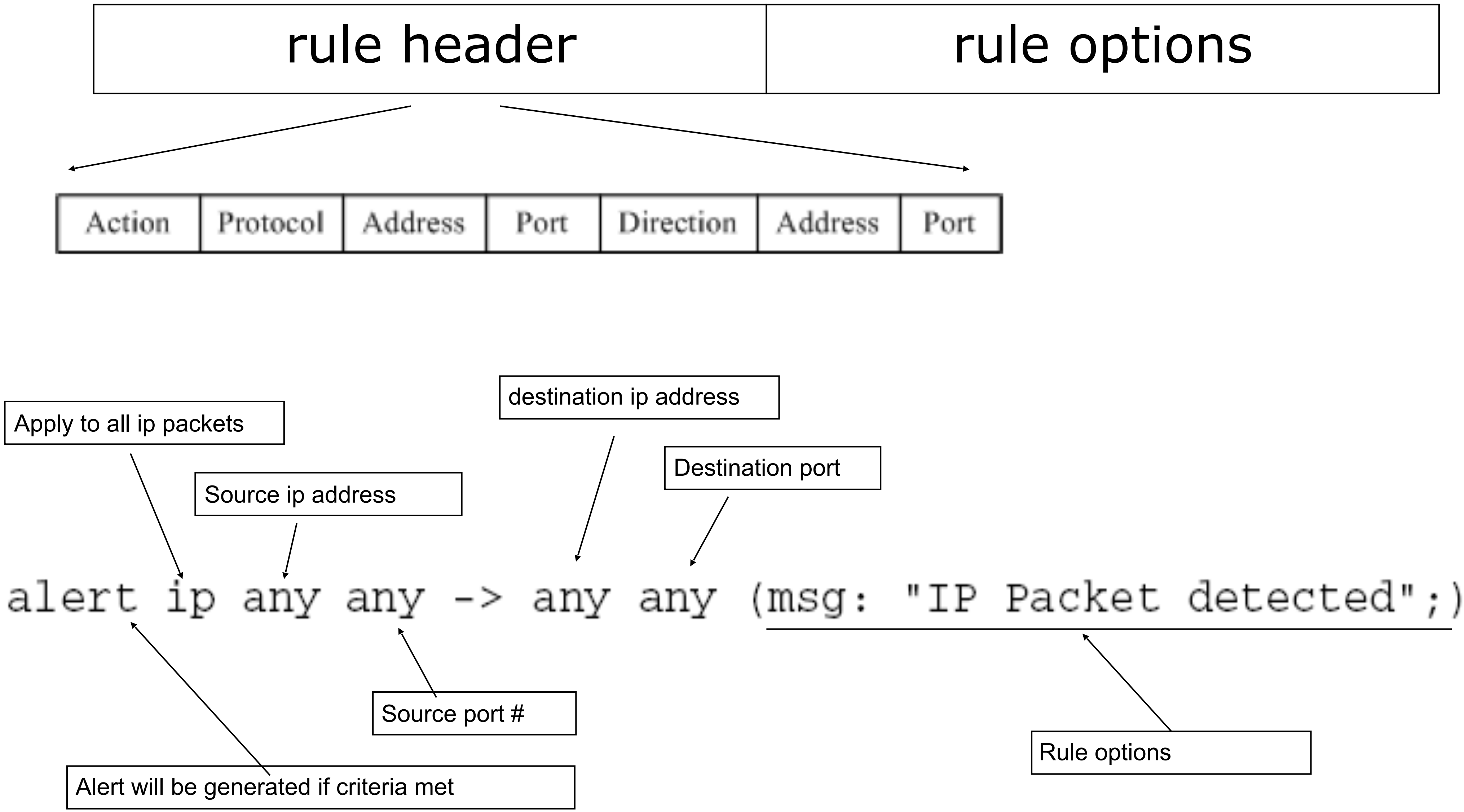


From: Rafeeq Ur Rehman, *Intrusion Detection Systems with Snort: Advanced IDS Techniques with Snort, Apache, MySQL, PHP, and ACID.*

Snort components

- Packet Decoder
  - input from Ethernet, SLIP, PPP…
- Preprocessor:
  - detect anomalies in packet headers
  - packet defragmentation
  - decode HTTP URI
  - reassemble TCP streams
- Detection Engine: applies rules to packets
- Logging and Alerting System
- Output Modules: alerts, log, other output

Snort detection rules



Additional examples

alert tcp any any -> 192.168.1.0/24 111
(content:"|00 01 86 a5|"; msg: "mountd access";)

alert tcp !192.168.1.0/24 any -> 192.168.1.0/24 111
(content: "|00 01 86 a5|"; msg: "external mountd access";)

! = negation operator in address

content - match content in packet

192.168.1.0/24 - addr from 192.168.1.1 to 192.168.1.255

https://www.snort.org/documents/snort-users-manual

Using a NIDS

- Plan your incident response process well before you install the system
- Know what you're looking for
- Make the system comprehensive
- Don't overreact to alarms
- If using a rules-based system, keep up with vulnerability reports