

Network Access Control

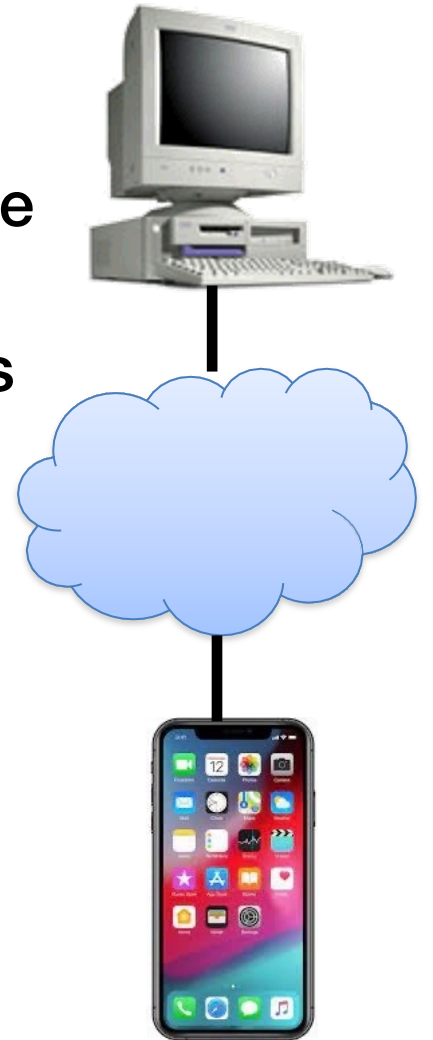


COS 316: Principles of Computer System Design
Lecture 16

Wyatt Lloyd & Rob Fish

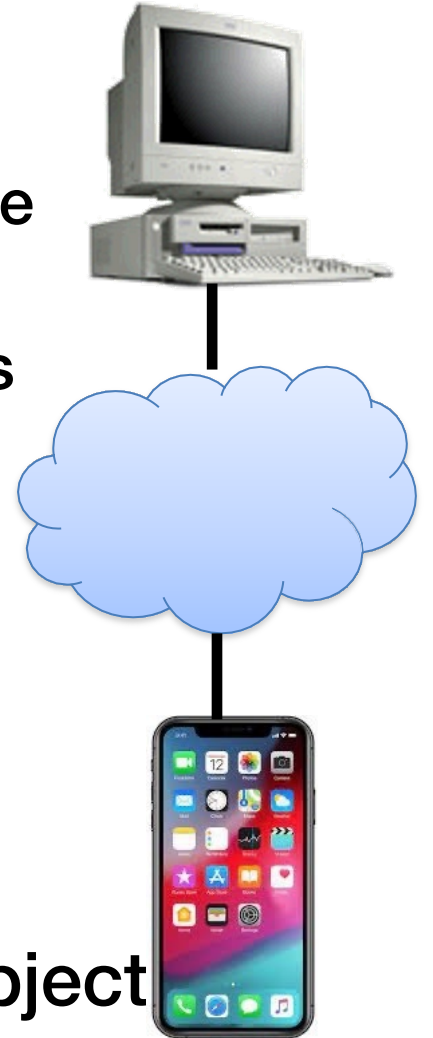
Controlling Which Packets Get Delivered

- **Objects:** the things being accessed
 - Services (possibly) running at the destination host machine
 - Identified by fields in the packet headers
 - E.g., destination IP address and TCP port number address
- **Subjects:** entity requesting access to an object

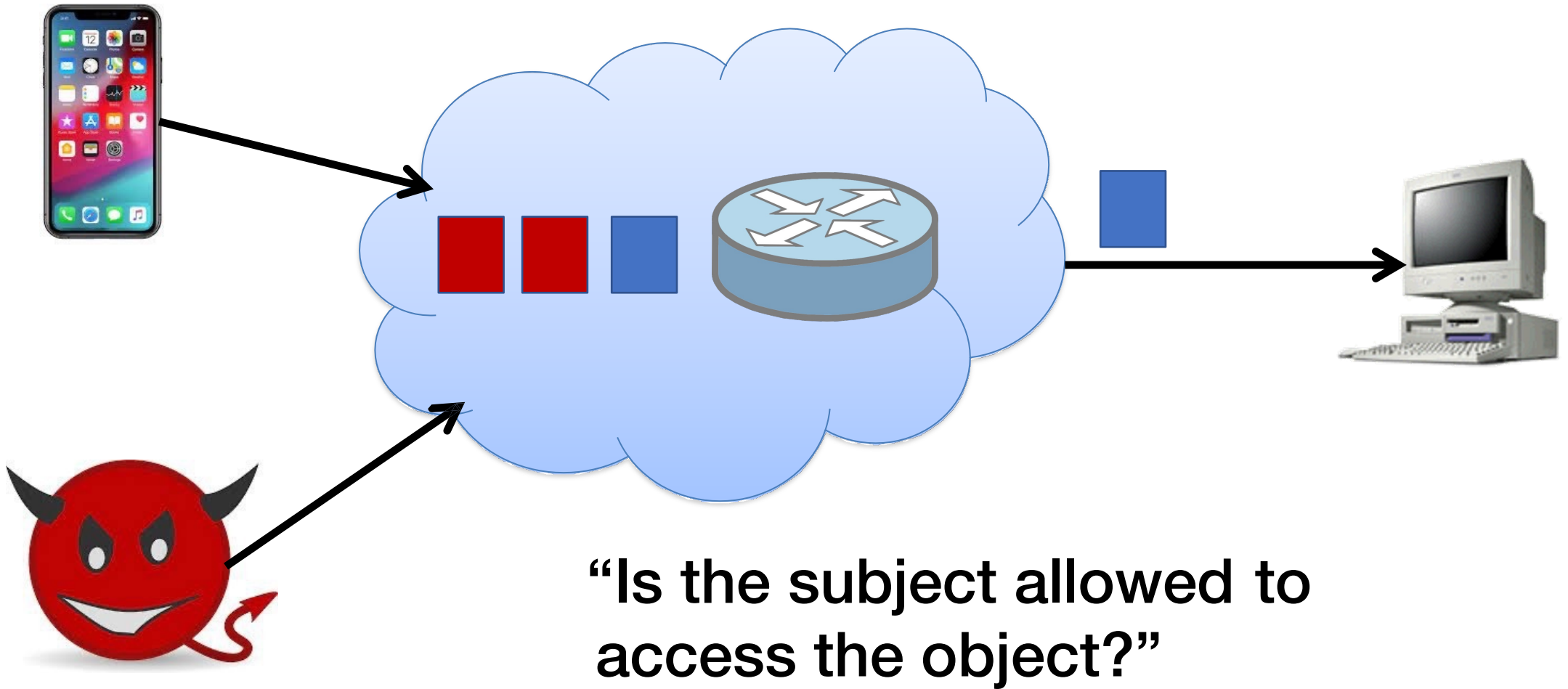


Controlling Which Packets Get Delivered

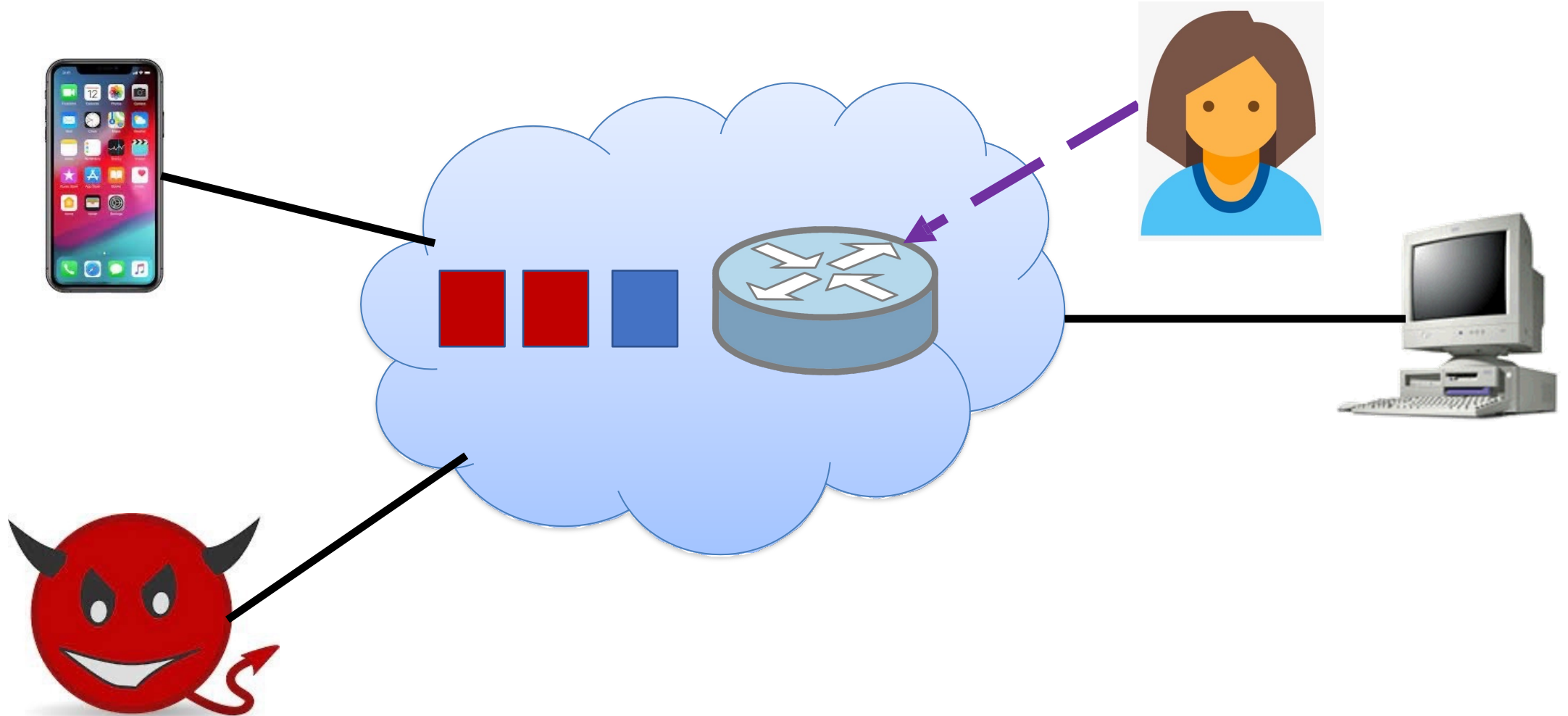
- **Objects:** the things being accessed
 - Services (possibly) running at the destination host machine
 - Identified by fields in the packet headers
 - E.g., destination IP address and TCP port number address
- **Subjects:** entity requesting access to an object
 - Sender of the packet on the source host machine
 - Identified by fields in the packet headers
 - E.g., source IP address, source TCP port number, ...
- **Authorization:** rules governing subject's access to object



The Guard Model

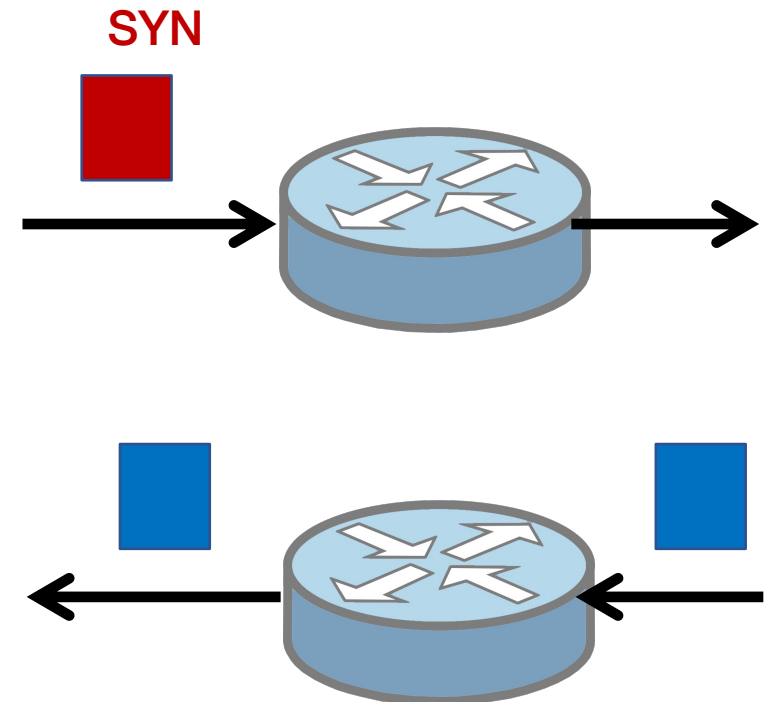


Network Administrator Sets the Policy



Policy Language: Access Control Rule

- An access control rule has two parts
 - Match: pattern on packet header fields and location
 - Action: permit (forward) or deny (drop)
- Block external initiation of a TCP connection
 - Match: external link, TCP protocol, TCP SYN flag
 - Action: deny
- Allow traffic from Princeton clients
 - Match: internal link, source IP in 128.112.*.*
 - Action: permit



Policy Language: Access Control Lists

- Access control list (ACL)
 - List of rules, possibly overlapping
 - Ordered list to disambiguate overlaps

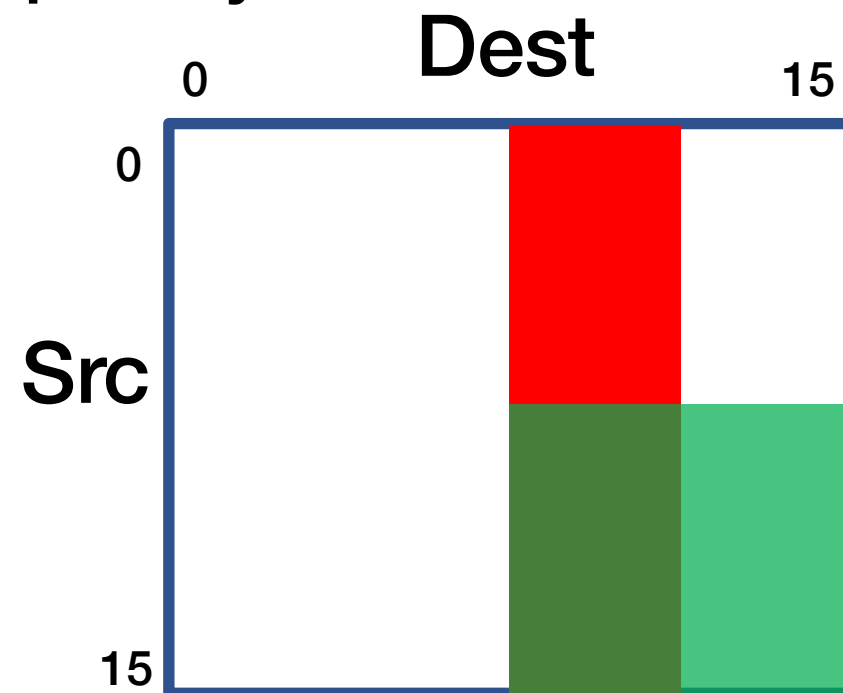
- Example:

Priority	Match	Action
1	Src=1.2.3.4, Dest=5.6.7.8	Deny
2	Dest=1.2.3.8, Dport=53	Allow
3	Dest=1.2.3.*	Deny
4	Src=1.2.3.7, Dport=100	Allow
5	Dport=100	Deny

Geometric Interpretation of Access Control List

- Overlapping shapes
 - Rules are multi-dimensional rectangles
 - Higher-priority rules on top of lower-priority
- Example with 4-bit addresses

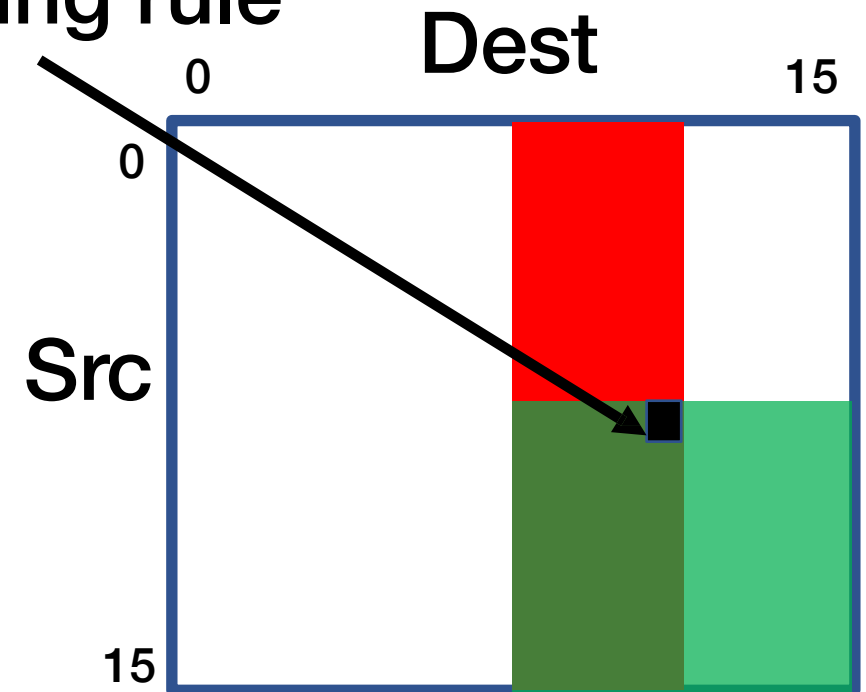
Pri	Match	Action
1	Src=1***, Dest=1***	Permit
2	Src=****, Dest=10**	Deny



Applying an Access Control List

- Classifying a packet
 - Packet header: Src=1000, Dest=1011
 - Find the highest-priority matching rule
- Apply the associated action

Pri	Match	Action
1	Src=1***, Dest=1***	Permit
2	Src=*****, Dest=10**	Deny



Simple Packet Classification Algorithm

- Classification problem
 - Given a packet (e.g., Src=1000, Dest=1011)
 - ... and an Access Control List
 - Find the highest-priority matching rule

- Simple algorithm
 - Scan rules in priority order
 - Stop after the first match
- *What is running time?*

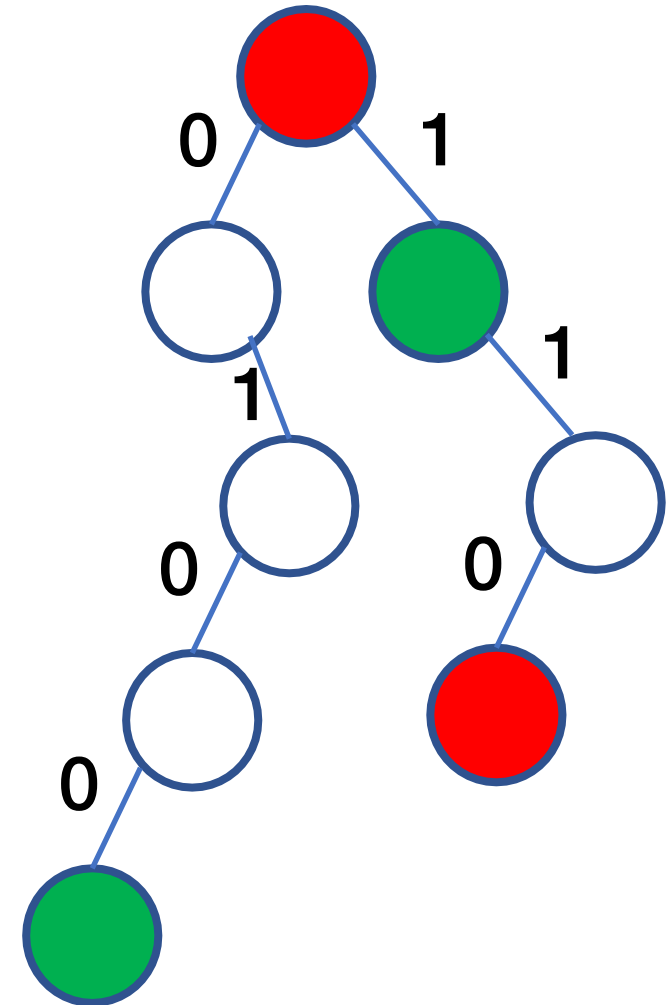
Pri	Match	Action
1	Src=1***, Dest=1***	Permit
2	Src=****, Dest=10**	Deny
3	Src=****, Dest=****	Permit

Special Case: One-Dimensional Prefix Matching

Pri	Match	Action
1	Dest=110*	Deny
2	Dest=0100	Permit
3	Dest=1***	Permit
4	Dest=****	Deny

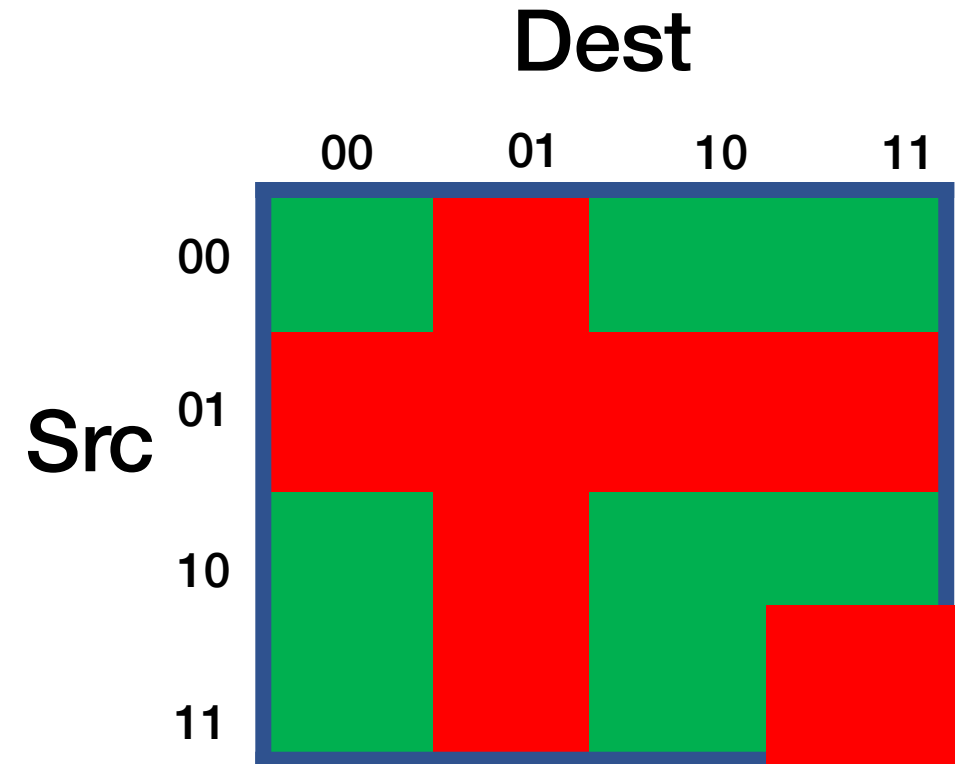
Longest-prefix match

Binary Trie



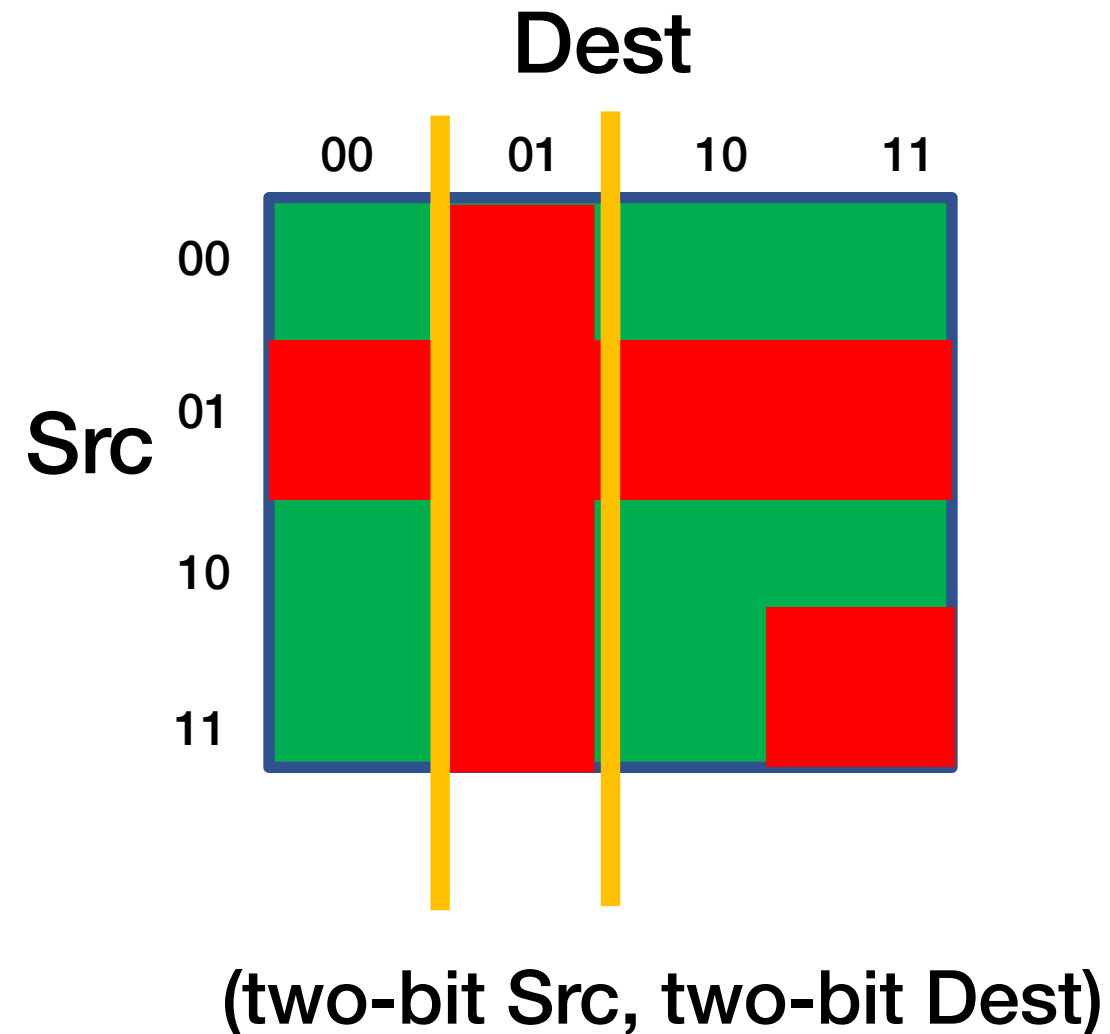
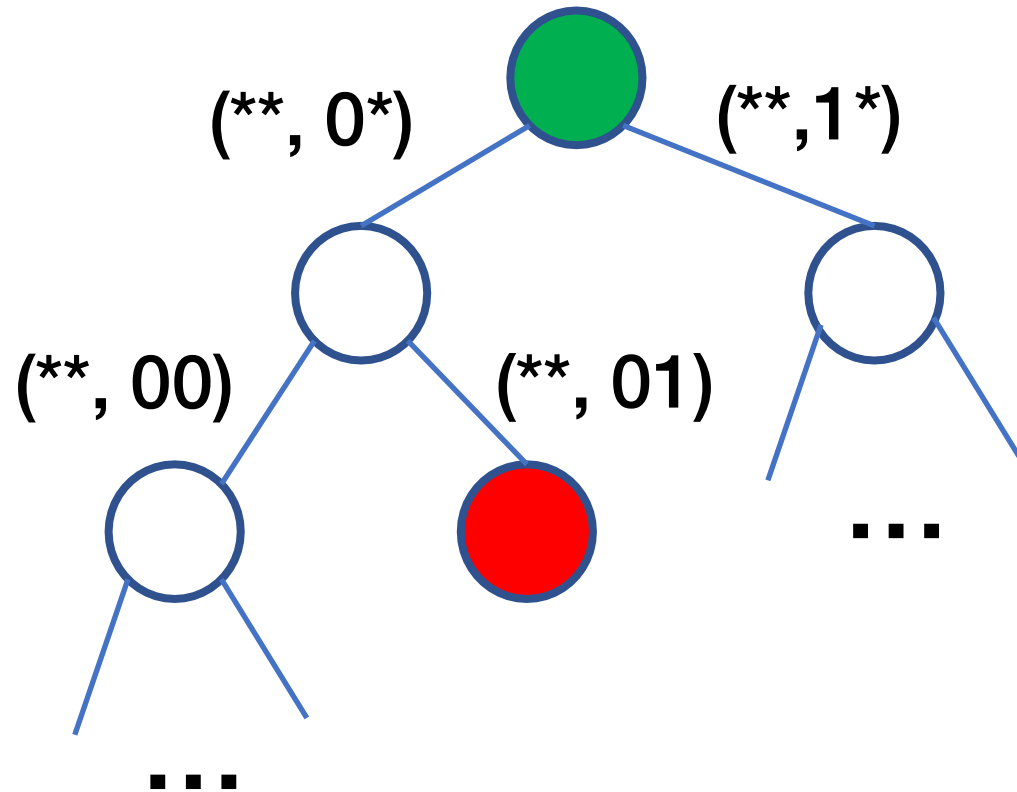
Multi-Dimensional Classification

Pri	Match	Action
1	Src=01, Dest=**	Deny
2	Src=**, Dest=01	Deny
3	Src=11, Dest=11	Deny
4	Src=**, Dest=**	Permit



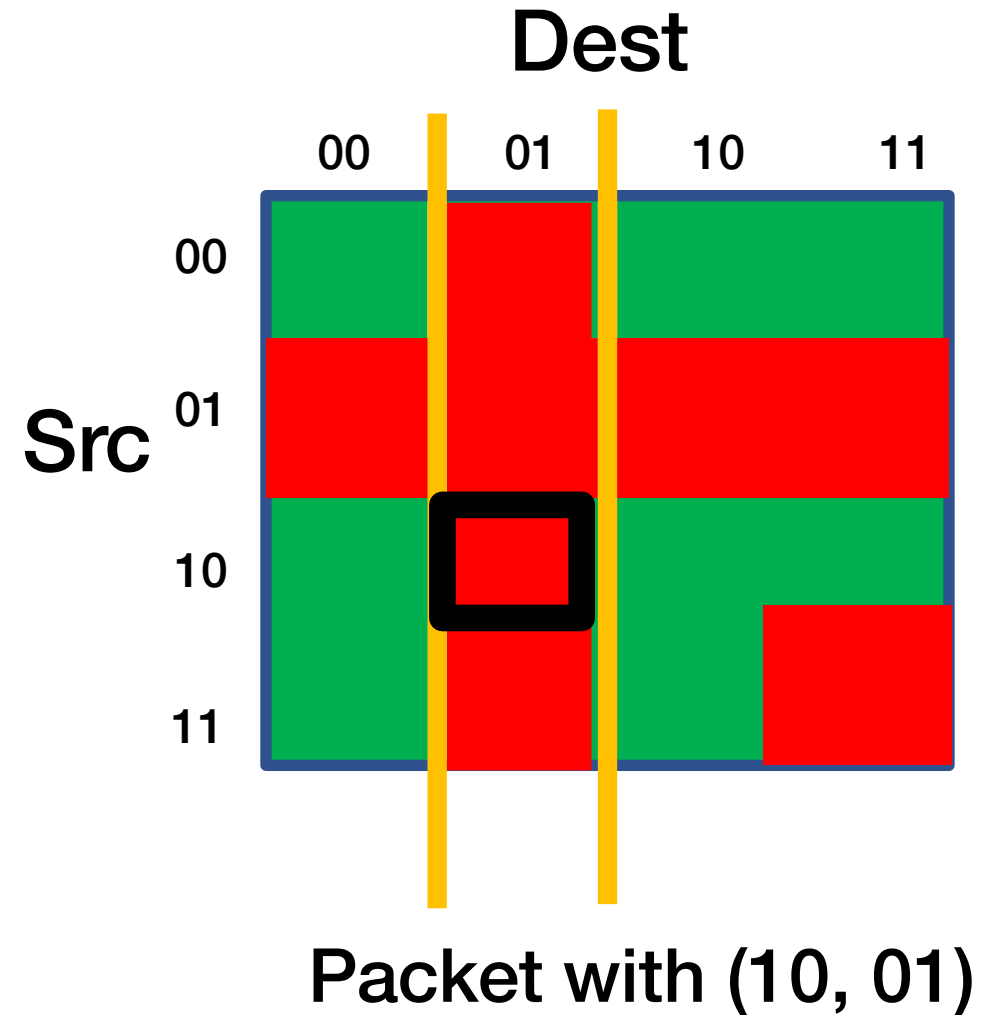
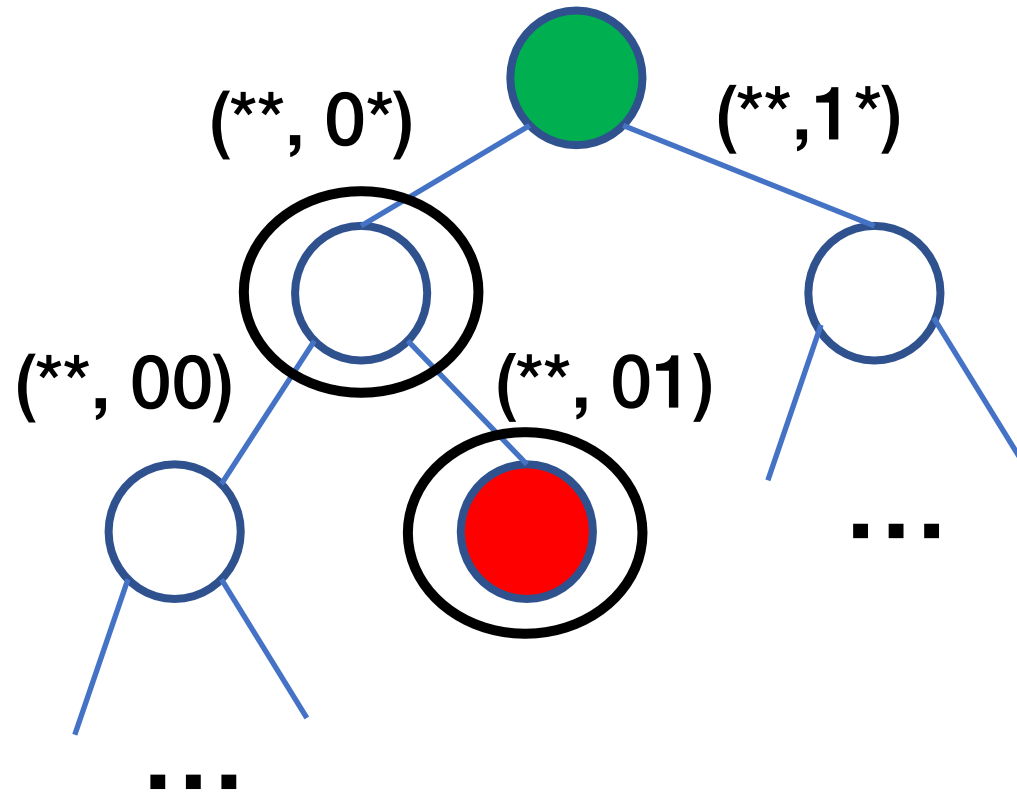
Multi-Dimensional Classification

- Build a compact classifier
 - By finding small “cuts”



Multi-Dimensional Classification

- Classify a packet
 - By traversing the trie



Packet Classification: CAM Hardware

- **Random Access Memory**
 - Given a memory address
 - ... return the data word stored at that address

00	b
01	a
10	d
11	c

- **Content-Addressable Memory**
 - Given some key
 - ... find the data word (if any) associated with the key

1010	b
0110	a
1110	d
0001	c

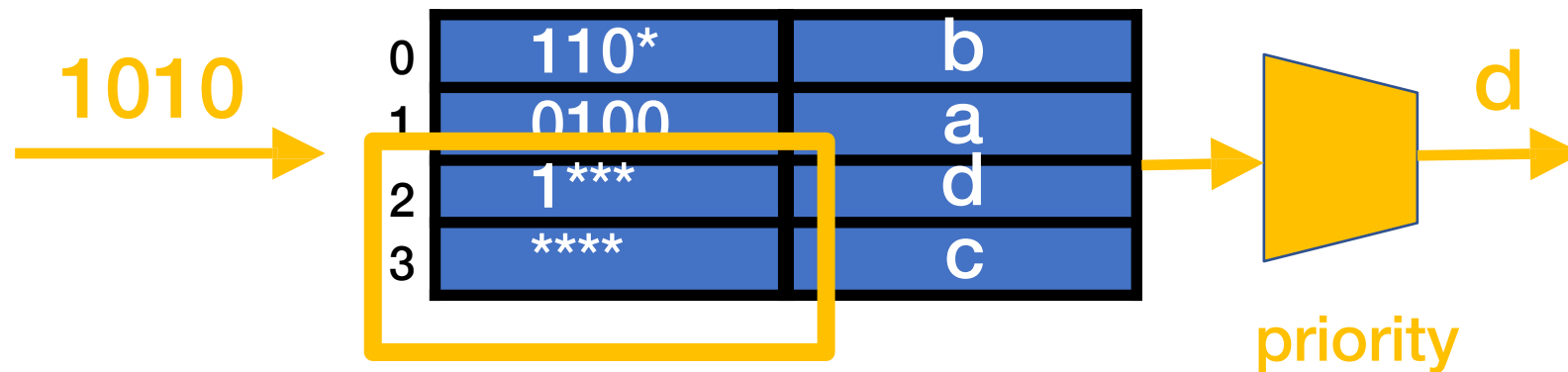
Packet Classification: Ternary CAM Hardware

- Ternary Content-Addressable Memory (TCAM)
 - Ternary: 0, 1, or * (wildcard)
 - Matching pattern can have wildcards
 - Entries in the TCAM in priority order

0	110*	b
1	0100	a
2	1***	d
3	****	c

Packet Classification: Ternary CAM Hardware

- Ternary Content-Addressable Memory (TCAM)
 - Ternary: 0, 1, or * (wildcard)
 - Matching pattern can have wildcards
 - Entries in the TCAM in priority order



Packet Classification in Practice

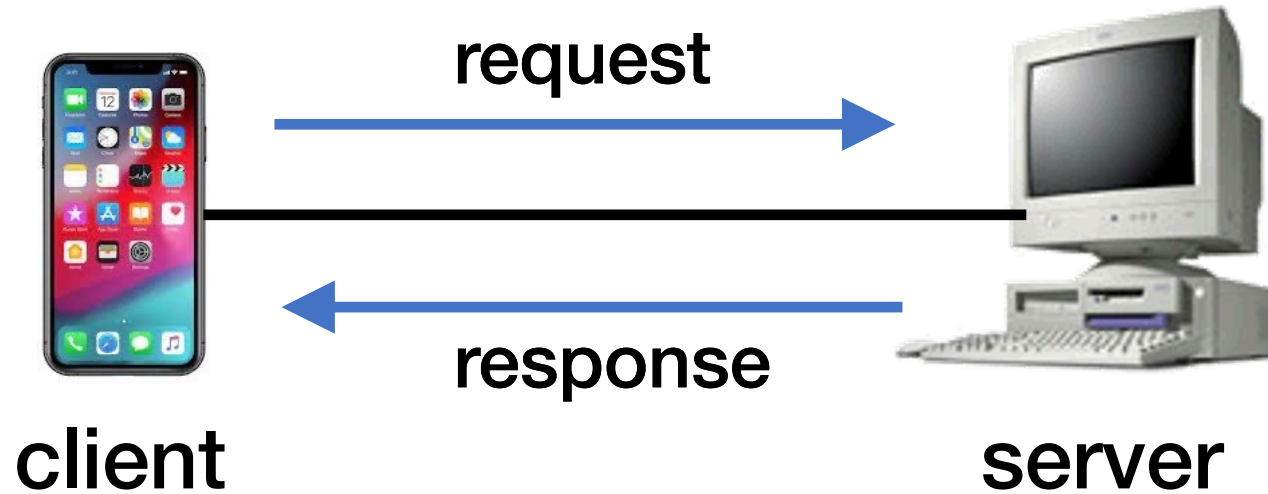
- **Software access control**
 - End-host network stack and software switches
 - Using algorithms for multi-dimensional packet classification
 - With optional caching of “popular” classification results
- **Hardware access control**
 - High-speed switches and network interface cards
 - Using Ternary Content Addressable Memory (TCAM)
 - With small TCAMs to reduce chip area and power consumption

Dynamic Access Control

- So far, we have discussed static ACLs
- Configured by a network administrator
 - Based on network administrator knowledge of (in)valid traffic
- More sophisticated policies are dynamic
 - Adapted to the ongoing traffic (e.g., stateful firewall, SYN cookies)
 - Adapted to the routing protocol (e.g., reverse path forwarding)



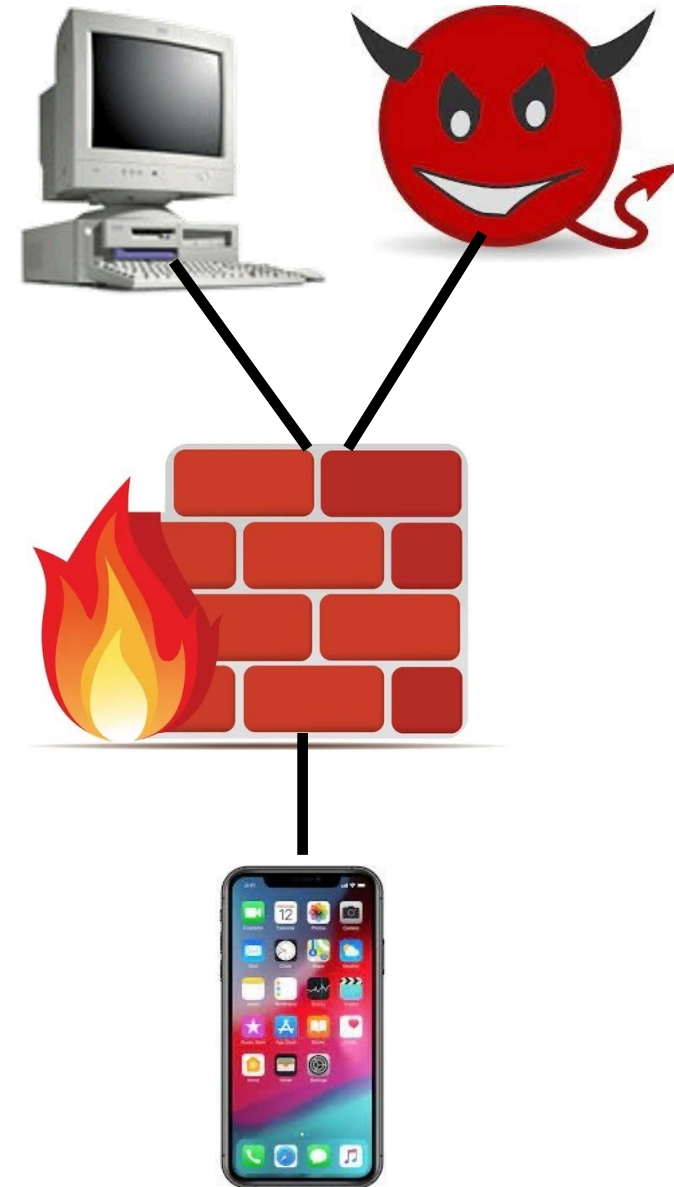
Internet Clients and Servers



- **Request-response protocols**
 - Client initiates communication by sending a request message
 - Server accepts the request and sends a response message

Stateful Firewall: Protecting Clients

- Most user devices act as a client
 - Sending DNS requests to look up domain names
 - Sending TCP SYN packets to start TCP connections
 - Sending HTTP requests to retrieve Web pages
- They should not receive unsolicited traffic
 - They should only receive response traffic
 - ... from requests they sent recently
- Stateful firewall
 - Remember recent client request traffic
 - ... and permit (only) the associated response traffic



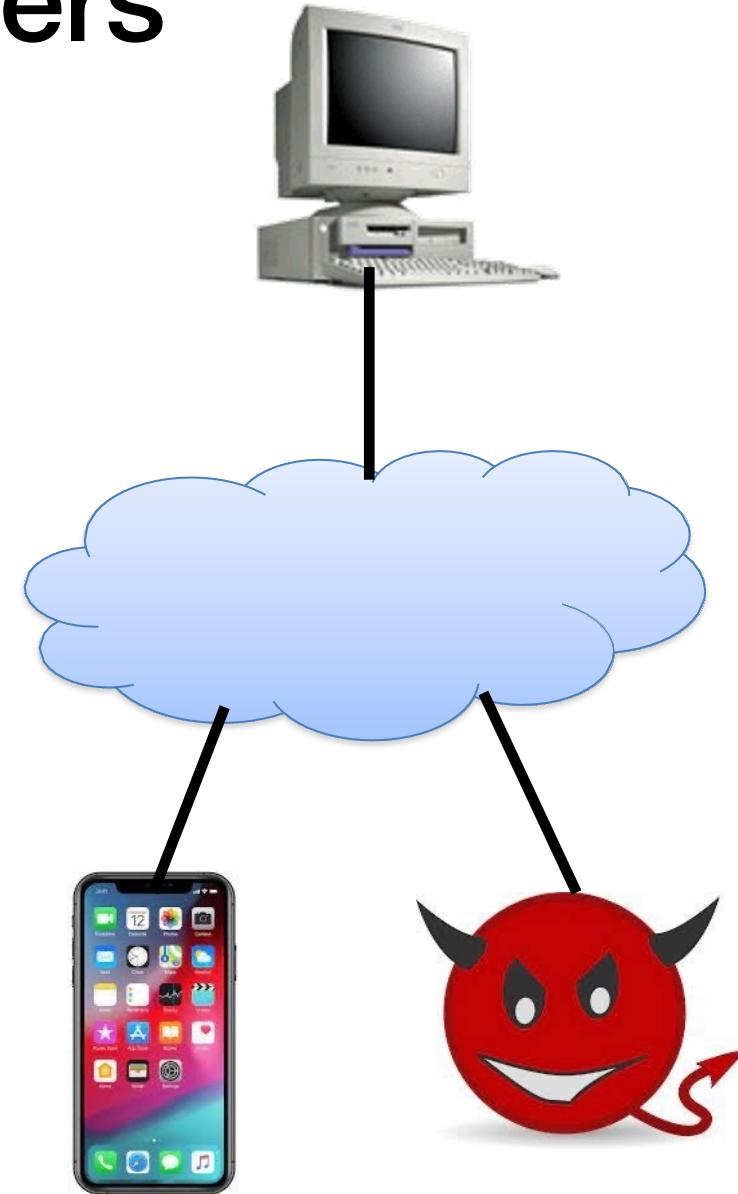
Stateful Firewall: Example



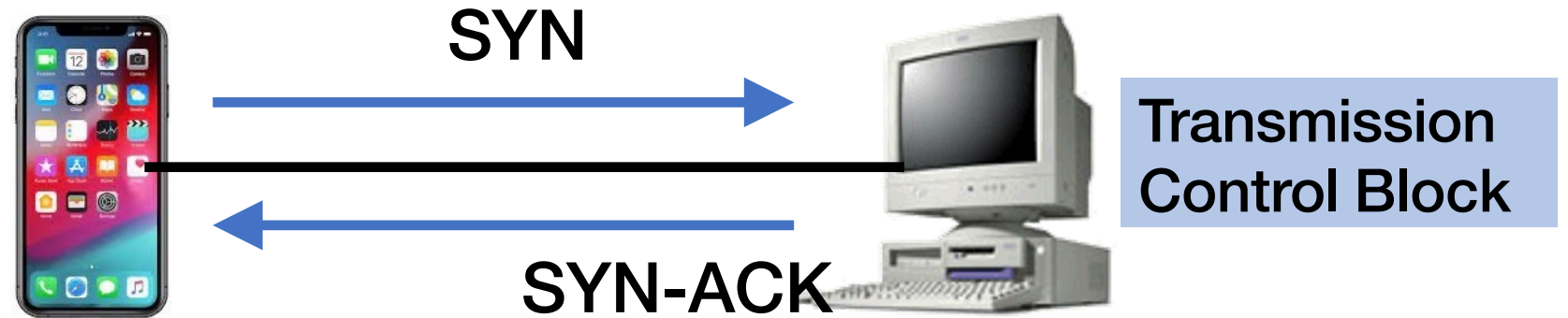
- By default, firewall denies all traffic destined to IP address 1.2.3.4
- Then, the client sends a packet to open a TCP connection to 5.6.7.8
- The firewall, on seeing the packet, adds a new “permit” rule
- ... allowing the return traffic from server 5.6.7.8 to client 1.2.3.4
- (Removing the rule when the connection ends or after a timeout)

SYN Cookies: Protecting Servers

- Denial-of-service attacks on servers
 - Malicious clients overloading the server
 - ... degrading performance of legit clients
- Challenging to prevent
 - Servers are supposed to receive traffic!
- Adversary's goal
 - Overwhelm the server
 - ... without investing much effort
 - Idea: asymmetric attack!

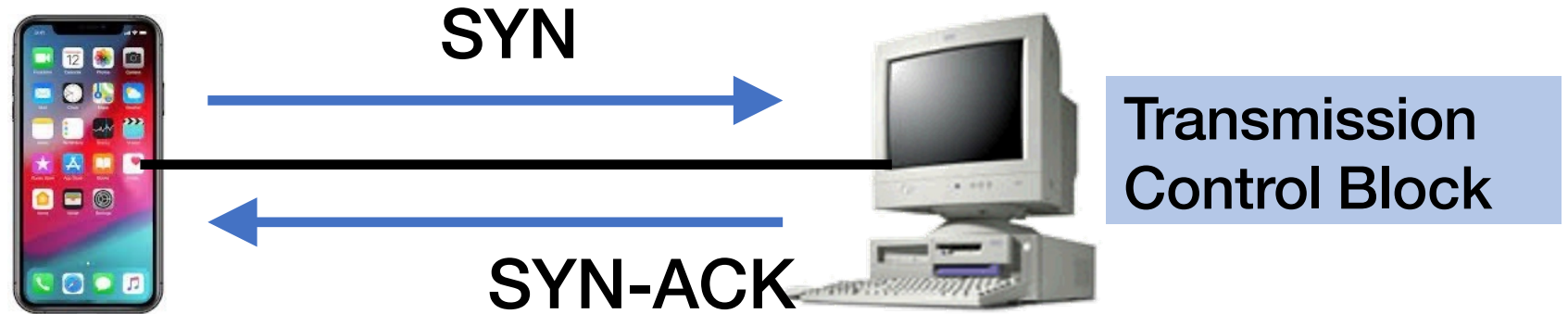


SYN Cookies: SYN Flooding Attacks



- **TCP handshake to start a connection**
 - Client sends a small SYN packet
 - Server allocates resources and sends a SYN-ACK
 - Client (supposedly) continues the communication

SYN Cookies: SYN Flooding Attacks



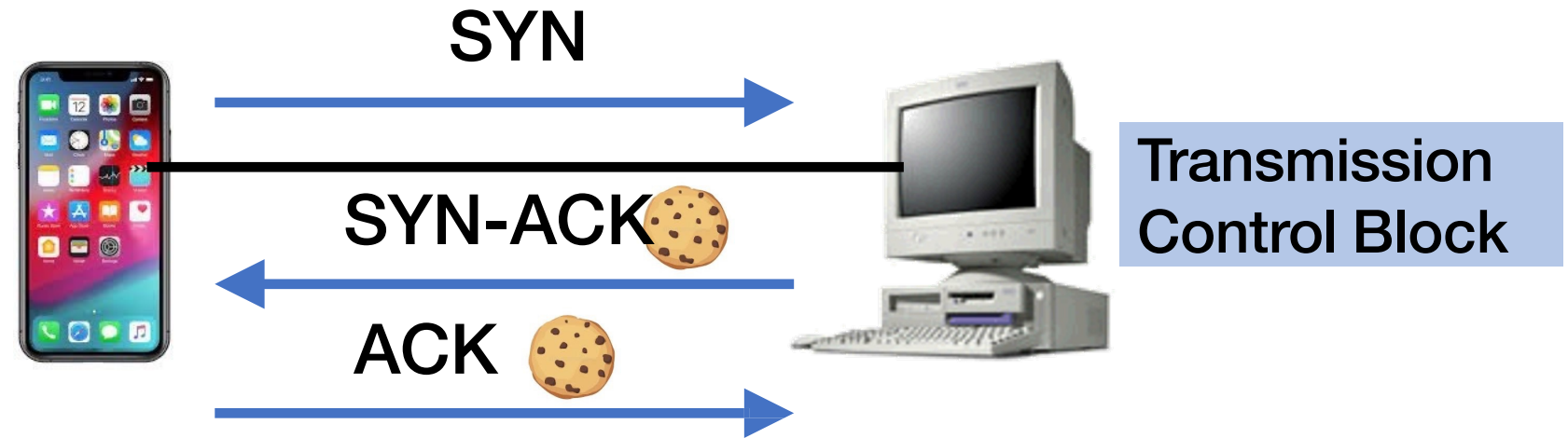
- **Asymmetric attack**

- Client sends a 40-byte SYN packet
- Server does a lot of work

- **Crafty adversary**

- Send from a spoofed source IP address (hard to trace!)
- Send from compromised hosts (very little overhead for adversary!)

SYN Cookies: Push the Work to the Client



- Server ensures the client has some “skin in the game”
 - Server puts a cryptographic “SYN cookie” in the SYN-ACK
 - Client must return the cookie in its ACK packets
 - Server verifies the cookie before dedicating resources
- Deny any ACK packets that fail the cookie check

Conclusions

- **Internet security is challenging**
 - Attackers can easily send unwanted traffic
 - ... that can compromise or overwhelm the destination computer
- **Access control is a crucial defense**
 - Blocking unwanted traffic based on packet header fields
 - Static access control policy when possible, dynamic when necessary
- **Enforcing access control lists**
 - Software algorithms for multi-dimensional packet classification
 - Ternary Content Addressable Memory (TCAMs)

