CS 268: Network Security

Kevin Lai

April 17, 2002

Motivation and Problem

- networks used for many critical services
  - financial transactions, journalism, business operations, etc.
- networks more open than ever before
  - global, ubiquitous Internet, wireless
- must prevent malicious users (examples?) from
  - reading data (privacy)
  - pretending to be someone else (authentication)
  - doing something without permission (authorization)
  - modifying transmitted data (integrity)
  - claiming they did not send a message (nonrepudiation)
  - denying service to other users (preventing denial-of-service)
- reduce key distribution problem
- detect a compromise by a malicious user (intrusion detection)

Privacy

- Prevent eavesdropper from reading message
- Encryption
  - Message M, key K, encryption algorithm E
    - $E(M,K) = \{M\}^K$
  - Given M^K, difficult to get M unless you have K
  - The more data encrypted and sent using K, the greater likelihood K can be deduced
    - K should be changed periodically
- Symmetric keys: DES, 3DES, blowfish, AES
  - $E(M,K) = \{M\}^K$, $D(\{M\}^K,K) = M$
  - K must be kept secret

Public/Asymmetric Keys

- K must be exchanged through secure medium
  - how to bootstrap?
- Asymmetric keys/public keys: DH, RSA, DSA
  - PK: Public Key, SK: Secret Key
  - E(M,SK) = $\{M\}^{SK}$, D($\{M\}^{SK}$,PK) = M
  - E(M,PK) = $\{M\}^{PK}$, D($\{M\}^{PK}$,SK) = M
  - DES 100 times faster than RSA in software
  - Typically, PK/SK used to exchange symmetric key, which is used for the conversation
  - PK can be exchanged “in the clear” (problem?)

Authentication

- Validate a mapping between two entities
  - alice@cs.berkeley.edu↔Alice
  - www.whitehouse.gov↔Whitehouse of USA
  - www.whitehouse.com↔entertainment provider (not Whitehouse of USA)
- Solutions
  - Passwords
  - Encryption
  - Biometrics

Integrity

- Verify that a message has not been modified
  - much stronger than checksum (difference?)
- Message digest/ characteristic function/ one-way hash: MD5, SHA
  - H(M) = h
  - h, H $\not\Rightarrow$ M (inversion resistance)
  - M $\not\Rightarrow$ M', s.t. H(M)=H(M')
  - $\not\Rightarrow$ M, M', s.t. H(M)=H(M') (collision resistance)
  - Additional mechanism to prevent attacker from also modifying hash
    - encrypt h, or
    - h = H(M,K), K is a secret key known by both sender and receiver

More Security

- Nonrepudiation
  - Prevent the sender from falsely denying he/she sent a message
  - Digital signatures
- Preventing denial of service
  - discussed later

More Security

- Intrusion Detection
  - described later
- Authorization (not discussed)
  - Determine if a user is allowed to do something
    - credit card authenticates a person
    - stores checks with the credit card company for spending limit authorization

Key Distribution Problem

- Many of the previous algorithms rely on keys
- How do two parties securely get keys to do privacy, authentication, etc.?
- Set up a secure connection using different key
  - How to bootstrap?
- Out-of-band key distribution
  - Floppy disk, piece of paper, telephone, etc.
  - High latency, wastes human time
- Must be done whenever key is compromised, entity is added, keys expire

Needham and Schroeder

- Addresses key distribution problem
- Reduces number of keys distributed out-of-band
- Assumes malicious user can read, modify, drop, and fabricate messages

Interactive Connection, Symmetric Key

1) A$\rightarrow$AS: A,B,I_{A1}

   to get CK from AS

   no encryption

2) AS $\rightarrow$A: $\{I_{A1},B,CK,\{CK,A\}^{KB}\}^{KA}$

   to send CK to A

   Encrypted with KA so only A can read it and so A knows it came from AS

   I_{A1} so that A knows this isn't a replay (why?)

   B so that A knows this isn't a man in middle attack (why?)

Interactive Connection, Symmetric Key

3) A→B: $\{CK,A\}^{KB}$

to send CK to B

encrypted with KB so that B knows it came from the AS and A is authenticated

4) B→A: $\{I_B\}^{CK}$

5) A→B: $\{I_B\text{-}1\}^{CK}$

so B can determine if 3) is a replay

Interactive Connection, Symmetric Key

- What if CK is compromised?
  - Attacker
    - listens to previous conversation between A and B
    - breaks CK eventually
    - spoofs A, sends copy of messages 3,4,5 to B
  - Add timestamp to messages:

  2) AS$\rightarrow$A: $\{I_{A1},B,CK,\{CK,A,TS\}^{KB}\}^{KA}$

  3) A$\rightarrow$B: $\{CK,A,TS\}^{KB}$

  B ignores if TS is too old
  - Need synchronized clock (why?)
    - How to secure clock synchronization protocol?

Interactive Connection, Asymmetric key

1) A$\rightarrow$AS: A,B

to get PKB from AS

2) AS$\rightarrow$A: $\{PKB,B\}^{SKAS}$

to send PKB to A

assume that A knows PKAS securely

encryption for integrity not privacy

B so that A knows 1) was good

3) A$\rightarrow$B: $\{I_A,A\}^{PKB}$

tells B that A wants to talk

Interactive Connection, Asymmetric key

4) B→AS: B,A

5) AS→B: $\{PKA,A\}^{SKAS}$

Same as 1) and 2)

6) B→A: $\{I_A,I_B\}^{PKA}$

Prevent replay from B to A

7) A→B: $\{I_B\}^{PKB}$

Prevent replay from A to B

Interactive Connection Comparison

- Messages sent
  - Symmetric key: 5, 3 with caching
  - Asymmetric key: 7, 3 with caching
  - Caching introduces vulnerabilities
    - key could have been compromised
- AS security
  - Symmetric key: must have privacy, integrity
  - Asymmetric key: needs only integrity

Advantages

- Resists some attacks
  - Eavesdropping
  - Replay
- Reduces number of persistent keys
  - Symmetric: n instead of n^2 (n: number of hosts)
  - Asymmetric: 2n + 2 instead of n^2
- Reduces out-of-band key distribution
  - Symmetric/asymmetric: n instead of n^2

Problems

- Authentication Server
  - Single point of failure
    - Could be compromised, crashed, overloaded
  - Must be securely administered
    - Must have administrator trusted by all principals
    - Adding principals requires contacting administrators → very slow
- Inter-domain communication
  - each domain has separate authentication server
  - Reverts to n^2 key distribution or
  - hierarchy of domains
    - parent domains must be trusted by child domains
  - Must go through administrator

Conclusion

- Systems derived from Needham-Schroeder
  - Kerberos
    - Popular in large centralized organizations
    - Centralized structure does not suit Internet
  - SSL
    - Used for secure TCP connections
- Key distribution is still a hard problem
  - many systems more vulnerable to key distribution attacks than crypto failure

```
The authenticity of host ‘host.domain.com (10.0.0.1)‘
can't be established.RSA key fingerprint is
be:3c:a3:8f:6d:70:32:78:e1:df:68:0f:ec:d2:f4:19.

Are you sure you want to continue connecting (yes/no)?
```

Denial of Service

- Huge problem in current Internet [MVS01]
  - Yahoo!, Amazon, eBay, CNN, Microsoft attacked
  - 12,000 attacks on 2,000 organizations in 3 weeks
  - some more that 600,000 packets/second
    - more than 192Mb/s
  - most documented perpetrators are determined teenagers using freely available tools
    - consider if the attacker is a large, well-funded group of professionals using secret tools
    - may have already happened
  - preventing deployment of critical applications
    - medical, energy, transportation

Problem: 0wning

- Attacker compromises a large number of hosts
  - 1M compromised hosts is plausible
- exploits security flaws in OS and applications
  - bugs, e.g., buffer overruns ("`strcpy(dest, src);`")
  - poor security policy, e.g., automatically executed email attachments
  - crypto, authentication systems do not prevent
  - firewalls do not prevent email viruses
- hosts usually have high bandwidth connections (e.g., DSL)

Problem: Attack

- Compromised hosts send TCP SYN packets to target
  - sent at max rate with spoofed source address
  - more sophisticated attacks possible
    - attack DNS, BGP
    - reflection
      - cause one non-compromised host to attack another
      - examples?
- Affect on target host
  - may crash or slow down drastically
  - connection to the Internet is saturated

Dealing with Attack

- distinguish attack from flash crowd (why?)
- prevent damage [M+01]
  - distinguish attack traffic from legitimate traffic
  - rate limit attack traffic
- stop attack
  - identify attacking machines
  - shutdown attacking machines
  - usually done manually, requires cooperation of ISPs, other users
- identify attacker
  - very difficult, except
  - usually brags/gloats about attack on IRC
  - also done manually, requires cooperation of ISPs, other users

Incomplete Solutions

- Fair queueing (why?)
- Integrated Services and Differentiated Services (why?)
- RSVP (why?)
- Quality of service mechanisms usually assume that users are selfish, but not malicious

Identifying Attacking Machines

- Defeat spoofed source addresses
- Does not stop or slow attack
- Egress filtering
  - a domain's border router drop outgoing packets which do not have a valid source address for that domain
  - if universal, could abolish spoofing (why isn't it universal?)
- IP Traceback [many proposals]
  - similar to DPS
  - routers probabilistically tag packets with an identifier
  - destination can infer path to true source after receiving enough packets

Aggregate Congestion Control [M+01]

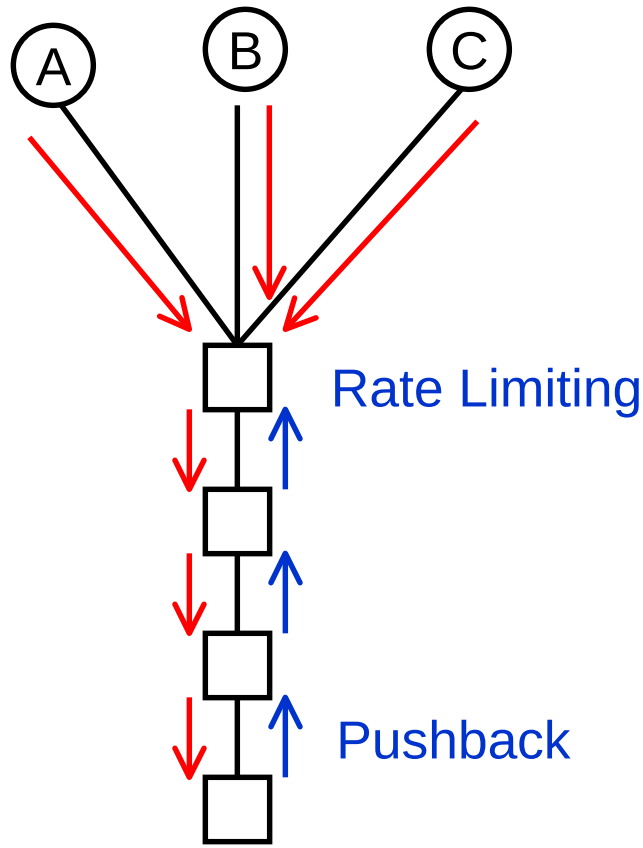
- goal: prevent damage from both attacks and flash crowds
- distinguish attack traffic from legitimate traffic
  - identify an aggregate of flows causing many drops
- limit aggregate
  - decide on bandwidth that limits drops
- convey decision to up stream routers
  - so up stream routers do not waste bandwidth delivering traffic that will be dropped

Distinguishing Aggregates

- Cluster together flows
- Too specific: does not affect drop rate (why?)
- Not specific enough: slow down legitimate traffic
- Cluster attributes: source/dest addr, source/dest port
- Examples
  - dest: cnn.com (+/-?)
  - dest: cnn.com/port 80 (+/-?)
  - dest: cnn.com/port 80, src: dosrus.com
- Clustering algorithm may have to be kept secret
- Current solutions use heuristics
  - open research problem

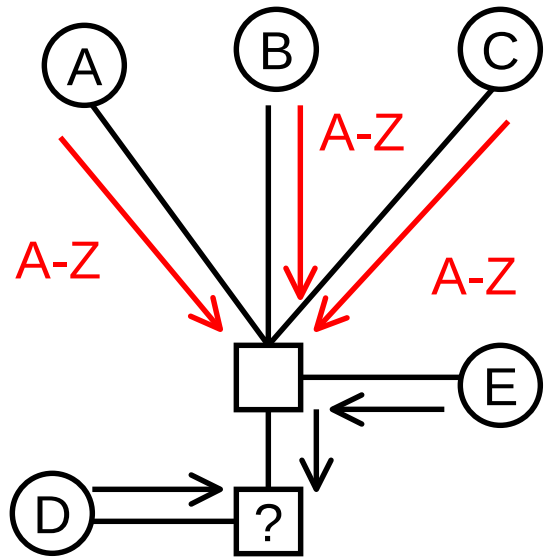
Pushback

- Convey information about high rate aggregate up stream
  - Why not necessary for flash crowd?
  - Why is it necessary for upstream routers to drop traffic?
  - Why do upstream routers need notification from downstream routers?



Pushback Issues

- Necessary if downstream router cannot identify aggregate
- Attack may still be too broad to distinguish
- Why would upstream routers trust downstream routers in different domains?



Conclusions

- Most significant problem in Internet today
- Traditional solutions ineffective
  - QoS, crytography, authentication
- Pushback provides general framework for solution
- Many problems remain

Network Intrusion Detection System (NIDS)

- Goal: automatically detect unauthorized access to hosts over the network
  - assume attacker has already compromised system
  - exploited inevitable flaws in system
    - bugs
    - compromised keys, passwords because of user mistakes
- maintain database of rules
  - e.g., “host X should never allow remote access”, “host Y should only be sent valid DNS queries”
- capture packets at border router and compare with database
- notify administrator in real time or automatically block intruder

Network Intrusion Detection Issues

- Why use NIDS in addition to firewall
  - NIDS doesn't block traffic, so it can protect hosts outside of firewall
  - Firewall doesn't prevent all forms of intrusion (e.g. email virus)
- Accuracy
  - rules are too general $\rightarrow$ too many false positives
  - rules are too specific $\rightarrow$ intruders undetected
- Fundamental rules
  - rules specific to application implementation $\rightarrow$ rule must change when application changes
  - application generic rules are difficult to formulate
  - e.g., interactive traffic can be characterized by distribution of human inter-character typing interval

- Little advantage for interactive communication
  - most people connect to only a fraction of the hosts in a domain $\rightarrow$ n is small
  - many hosts share same keys $\rightarrow$ n is small
  - user changes set of hosts with distinct keys infrequently
    - with PK, user can collect all PKs (n) and copy them to all hosts (n) $\rightarrow$ 2n key distribution instead of n^2