Internet Routing Security Issues

Z. Morley Mao

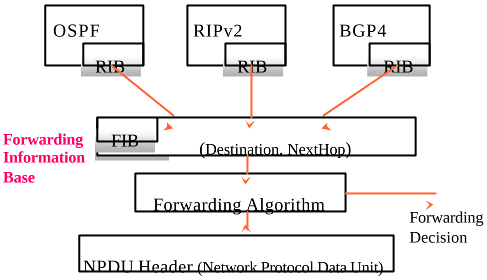
Lecture 4

Jan 21, 2003

Outline

- Recap of last lecture, questions?
- Comparison of routing protocol design
- OSPF security issues
- Discussion of research project suggestion in routing security

Routing Protocol Framework - Information Model



OSPF RIB

RIPv2 RIB

BGP4 RIB

Forwarding Information Base

FIB (Destination, NextHop)

Forwarding Algorithm

Forwarding Decision

NPDU Header (Network Protocol Data Unit)

RIB vs. FIB

- RIB: Routing Information Base
  - holds all routing information received from routing peers
    - Adj-RIBs-In, the Loc-RIB, and the Adj-RIBs-Out
    - routes that will be used by the local BGP speaker must be present in the Loc-RIB
    - routes that are received from other BGP speakers are present in the Adj-RIBs-In
- FIB: Forwarding Information Base
  - minimum amount of information necessary to make a forwarding decision on a particular packet
  - Typically: network prefix and next hop information
  - Contains unique paths, no secondary paths
  - Size of the FIB influences the speed of forwarding due to longest prefix lookup

Considerations in validating the path in routing protocols

draft-white-pathconsiderations-00.txt

- Path vector protocol participant cannot verify
  - whether the path a packet takes to its destination corresponds to the path advertised by the routing protocol
  - whether the chosen path is in accordance with the policies of other ASes.
- This due to
  - path vector routing protocols abstract information about intra-AS routing decisions
  - ASes can remove routes form the routing systems, this may prevent another AS from enforcing its own policy

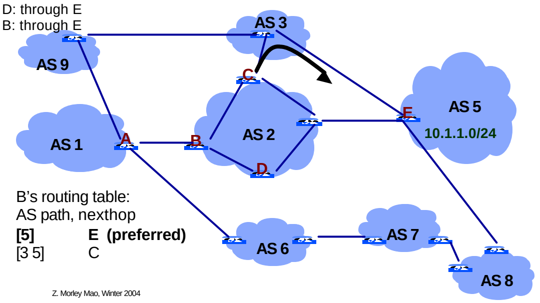
Validity of a path

1. Does a path from the advertising router to the destination advertised actually exist?
2. Does the path advertised fall within the policies of the route's originator and all intermediate autonomous systems?
3. Is the advertising router authorized to advertise a path to the destination?

- 2 and 3 cannot be verified in a distance or path vector protocol

Example 1
The advertised path may not fall within the policies of the receiver

E: local path
C: through AS 3
D: through E
B: through E



B’s routing table:
AS path, nexthop

[5]	**E (preferred)**
[3 5]	C

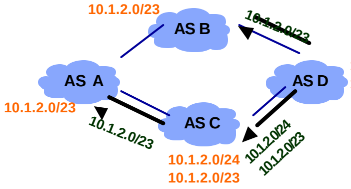
Some subtleties here

- BGP forwarding information looks like this:
  - Prefix and **nexthop**
  - Nexthop is the IP address of the nexthop router for forwarding traffic
  - You must have the IGP route to the nexthop for the route to be usable
- When B forwards traffic, it goes through C to reach E – the nexthop of the path
- C’s forwarding table is inconsistent with B
  - It prefers AS path [2 3 5]

Why can this happen?

- Intra-AS configuration of an AS can cause packets to follow a path inconsistent with advertised path
- Internal inconsistency in routing decisions within an AS
  - Path vector routing depends interior routing protocols
- Other examples: route reflection
- Any lesson here?
  - Guarantee the consistency of routes for all routers within an AS

Example 2
Advertising router may not be authorized to advertised a path to the destination



D prefers
10.1.2.0/24 from C (more secure)
10.1.3.0/24 from B

- A does not receive 10.1.2.0/24 from C
- A’s choice of [B D] overrides D’s implicit policy of only accepting this traffic from C
- This is due to removal of information from the routing system
- Lack of information does NOT mean lack of authorization to transit a path

How can routing information be “deleted”?

- Filtering based on prefix length
- Filtering based on the presence of supernets
- Filtering based on receiver
  - Doesn’t want to transit traffic for a peer
- Very prevalent especially between peers or inside Internet core

Comparison

Type of protocol	Advantages	Limitations
Link-state	Fast convergence Low churn/major event High visibility	Lack of scalability, isolation
Distance-vector	Isolation, Scalability, simplicity	Loops, count to infinity, slow convergence, little visibility, high churn
Path-vector	No routing loops, No count to infinity, Scalability, reasonable visibility	No isolation, Slow convergence, High churn

OSPF

- Link State routing protocol (RFC1583)
- Routers are organized in domains and areas
- Hello message for neighbor acquisition
- Link State information are flooded through the whole area
- A topology database is maintained by every router

Important LSA fields

- Advertising router ID (originator)
- Advertised link or network ID
- Sequence number [0x80000001,0x7fffffff]
- Age [0, 60 minutes]

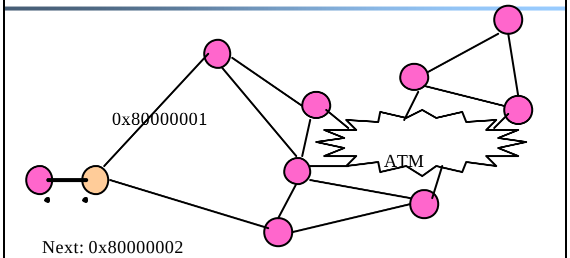
When to Originate a LSA?

- Upon link state changes, or
- Upon timer expiration

Questions to Ask:

- How do you know one LSA is fresher than the other?
- An LSA originated by you will be received by every router; will you receive the LSA originated by you?
- Will the sequence number wrap-around cause any problem? (i.e., == 0x7fffffff)
- Age ==> 1 hour

Sequence #: old vs. new LSAs

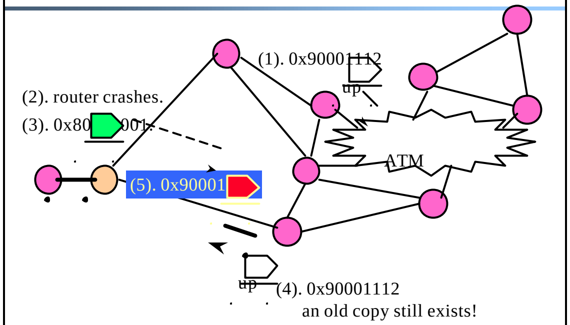


0x80000001

Next: 0x80000002

Only accept LSAs with newer/larger Seq#.

Sequence# & Self-Stabilization



(1). 0x90001112

(2). router crashes.

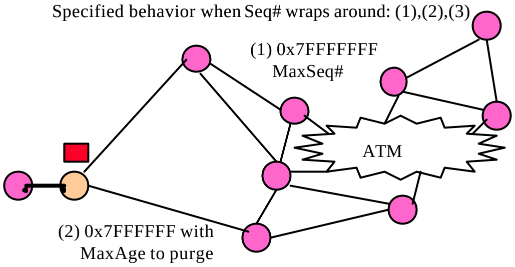
(3). 0x80 001.

(5). 0x90001

(4). 0x90001112
an old copy still exists!

Flushing via Premature Aging

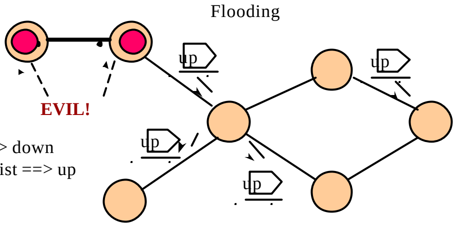
Specified behavior when Seq# wraps around: (1),(2),(3)



(2) 0x7FFFFFF with MaxAge to purge this entry.

(3) 0x80000001.

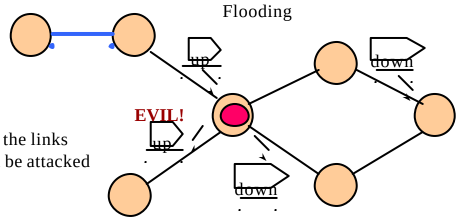
Attack the Routing Infrastructure (Vicious Advertising Routers)



1. up ==> down
2. not exist ==> up

Impact varies depending on how critical the link is to the world!

Attack the Routing Infrastructure (Vicious Intermediate Routers)

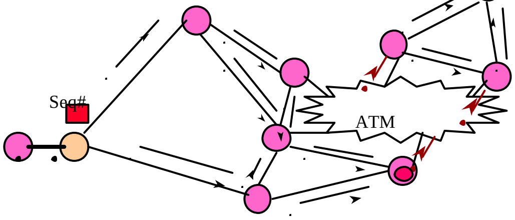


All the links can be attacked

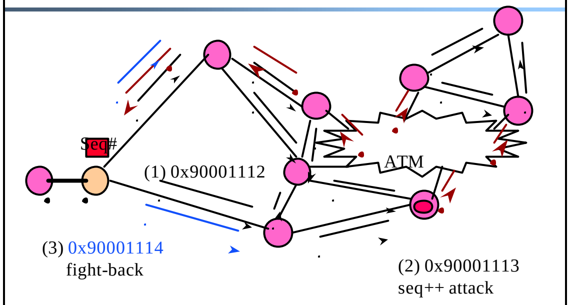
Authentication, please come to the rescue!

Exchanging without LSA Signature?

If attackers can just change the content of LSAs without being detected, the routers must use all LSAs with care!



Fight-Back - Originator Reaction

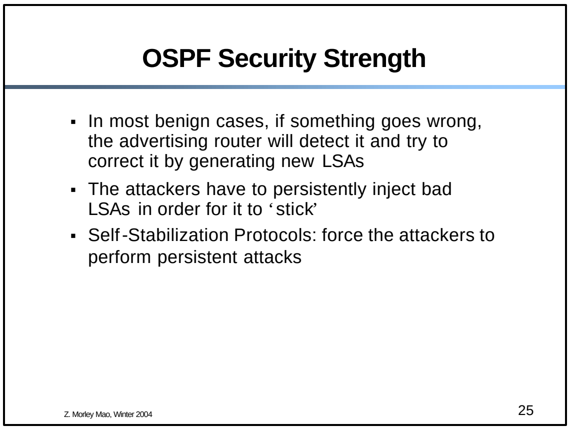


(3) 0x90001114 fight-back

(2) 0x90001113 seq++ attack

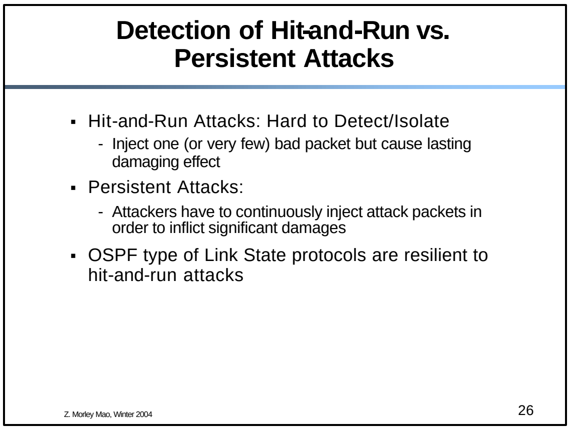
Signature - How Critical?

- Observations:
  - Prolonged fight-back will not happen in real attacks
  - What's preventing the attacker from using LS_seq=MaxSeq?
- Can you prevent false LSA without signature?
- Can you determine who did it after you realize that you've been fooled without signature?
- What needs to be signed by whom anyway?



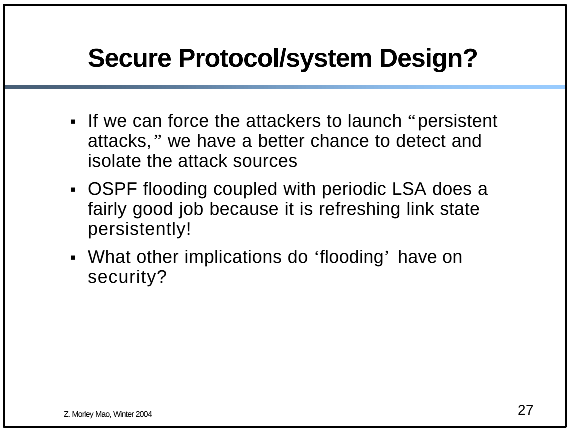
OSPF Security Strength

- In most benign cases, if something goes wrong, the advertising router will detect it and try to correct it by generating new LSAs
- The attackers have to persistently inject bad LSAs in order for it to ‘stick’
- Self-Stabilization Protocols: force the attackers to perform persistent attacks



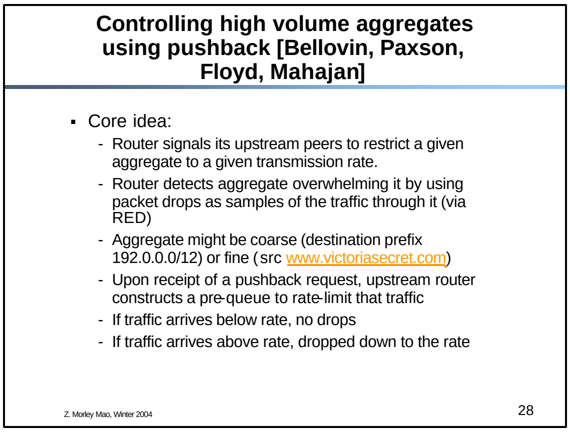
Detection of Hit-and-Run vs. Persistent Attacks

- Hit-and-Run Attacks: Hard to Detect/Isolate
  - Inject one (or very few) bad packet but cause lasting damaging effect
- Persistent Attacks:
  - Attackers have to continuously inject attack packets in order to inflict significant damages
- OSPF type of Link State protocols are resilient to hit-and-run attacks



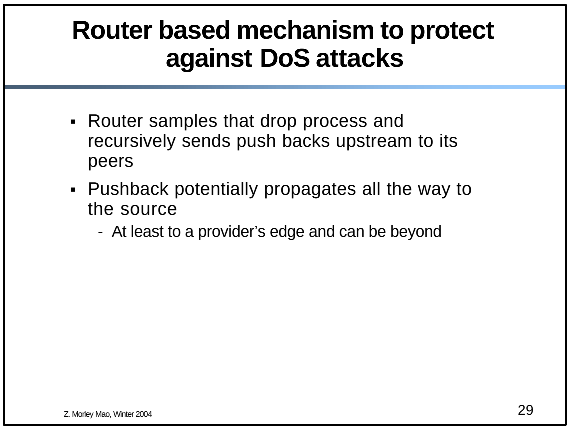
Secure Protocol/system Design?

- If we can force the attackers to launch “persistent attacks,” we have a better chance to detect and isolate the attack sources
- OSPF flooding coupled with periodic LSA does a fairly good job because it is refreshing link state persistently!
- What other implications do ‘flooding’ have on security?



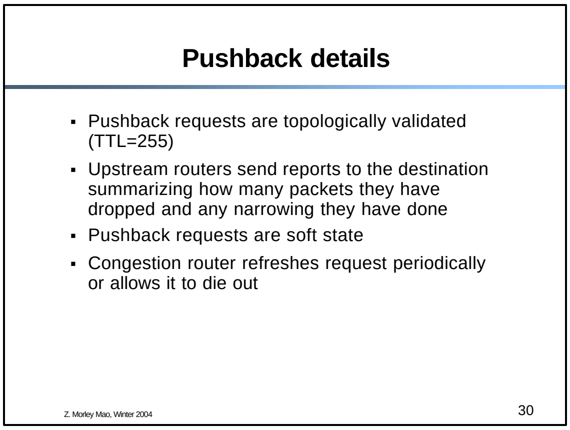
Controlling high volume aggregates using pushback [Bellovin, Paxson, Floyd, Mahajan]

- Core idea:
  - Router signals its upstream peers to restrict a given aggregate to a given transmission rate.
  - Router detects aggregate overwhelming it by using packet drops as samples of the traffic through it (via RED)
  - Aggregate might be coarse (destination prefix 192.0.0.0/12) or fine (src www.victoriasecret.com)
  - Upon receipt of a pushback request, upstream router constructs a pre-queue to rate-limit that traffic
  - If traffic arrives below rate, no drops
  - If traffic arrives above rate, dropped down to the rate



Router based mechanism to protect against DoS attacks

- Router samples that drop process and recursively sends push backs upstream to its peers
- Pushback potentially propagates all the way to the source
  - At least to a provider’s edge and can be beyond



Pushback details

- Pushback requests are topologically validated (TTL=255)
- Upstream routers send reports to the destination summarizing how many packets they have dropped and any narrowing they have done
- Pushback requests are soft state
- Congestion router refreshes request periodically or allows it to die out

Open questions:

- General mechanism for controlling high-bandwidth aggregates, e.g., flash crowds
- It does not protect against DDoS attacks with diverse sources
- Trust issues across networks
- What are the time constants?
- How does it interact with traffic management services?

Research project suggestions

- Analyze a new attack against routing protocols and devise a defense mechanism
  - Route flap damping attack
- Design router primitives to defend against DDoS, Worm, infrastructure attacks
  - Push back for DDoS
- How to exploit topology information to launch routing attacks
  - Variations of link-cutting attacks
- Attack detection
  - Exchange of information among ISPs
  - Signature, behavior based
  - Routing protocol analyzers (Bro)
- Intradomain topology design considerations
  - Route reflector vs. AS confederations or hybrid
  - Robustness, ease of configuration, security/trust

Projects ideas continued…

- Dynamic installation of route filters to protect against DDoS attacks