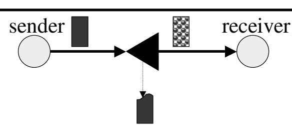
Securing Information Transmission by Redundancy

Jun Li Peter Reiher Gerald Popek
Computer Science Department
UCLA

Outline

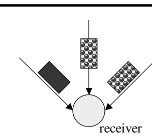
- Interruption threats are hard to counter
- Redundant transmission makes interruptions harder
- But redundant transmission is not as easy as using redundancy in other areas
- An experimental system – Revere
- Future Work
- Conclusion

Problem



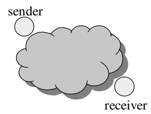
- Transmission is subject to interruption threats
  - Path interruption
  - Data interruption
- Conventional approaches won't help
  - Encrypted/signed message can still be interrupted
- Acknowledgement won't help either
  - The acknowledgement itself is subject to interruption
  - Retransmission means possibly failing again
- So?

Using Redundancy To Counter Interruptions



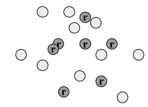
- Don't use a single path
  - Any point on the single path is a point of failure
- Use redundancy to secure transmission
  - Only parallel redundancy considered here
    - A node is expected to receive multiple copies of one message
    - Successful if at least one copy is authentically received
- Redundancy has been widely used in other areas
  - High availability storage
  - Replicated execution
  - Mapping one web site to several server machines
  - And many others
- But …

Transmission Redundancy Is Not Easy



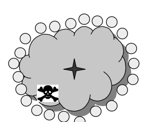
- Redundant transmission is tough
  - Discovering disjoint paths is difficult
    - Routing is transparent to applications
  - Using disjoint paths is difficult
  - May not exist at all
- Can try to be as disjoint as possible nevertheless
  - An attacker has to find a choke point or break multiple points

Redundancy For Large-Scale Situations



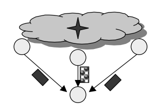
- High scale adds further problems
- Structure for redundant transmission can only be built in distributed fashion
- Routing infrastructure itself is not secure enough
  - A router on one path may be fooled by a router on another path
- Tradeoff between resource usage and redundancy degree
- Heterogeneity
  - Each node may have different requirement on security assurance, different transmission characteristics, different platform, etc.

Revere



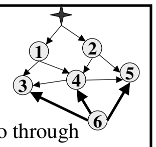
- Goal: disseminate security updates to large number of machines
- Assume a trusted dissemination center
- Security updates
  - Small size but critical information
  - Examples:
    - New virus signature
    - New intrusion detection signature
    - CRL (certificate revocation lists)
    - Offending characteristics for a firewall to monitor

Revere Structure



- Cannot use acknowledgement or negative acknowledgement
- Employ redundancy to have multiple copies sent to each node
  - Each node can also forward security updates to others
- A node can contact multiple repository nodes for missed updates

Revere Structure Formation



- Each node maintains several path vectors
  - A path vector = <an ordered list of nodes to go through to reach this node from center, latency>
  - For example,

❸	<❶, 1sec>	<❷❹, 1.5 sec>
❹	<❶, 1sec>	<❷, 0.5 sec>
❺	<❶❹, 1sec>	<❷, 10 sec>

- When a new node joins
  - Contact several existing Revere nodes
  - Compare their path vectors to find parent nodes
    - Efficiency and resiliency as two most significant factors
    - For example, after contacting ❸❹❺, ❻ choose ❹ and ❸ and setting up two initial path vectors:

❻	<❷❹, 1.5sec>	<❶❸, 2sec>

- Maintain the structure
  - Heartbeat messages

Future Research

- Further understanding of large-scale redundancy
- Security of the transmission structure itself
- Theoretical aspects such as resiliency evaluation
- Deployment in Internet

Conclusion

- Conventional security approaches and transmission primitives are not adequate in counteracting interruption threats
- Using redundancy is promising
- But effective use of redundancy is challenging

Questions

- Contact information
  - Jun Li, lijun@cs.ucla.edu
  - Peter Reiher, reiher@cs.ucla.edu
  - Gerald Popek, popek@cs.ucla.edu