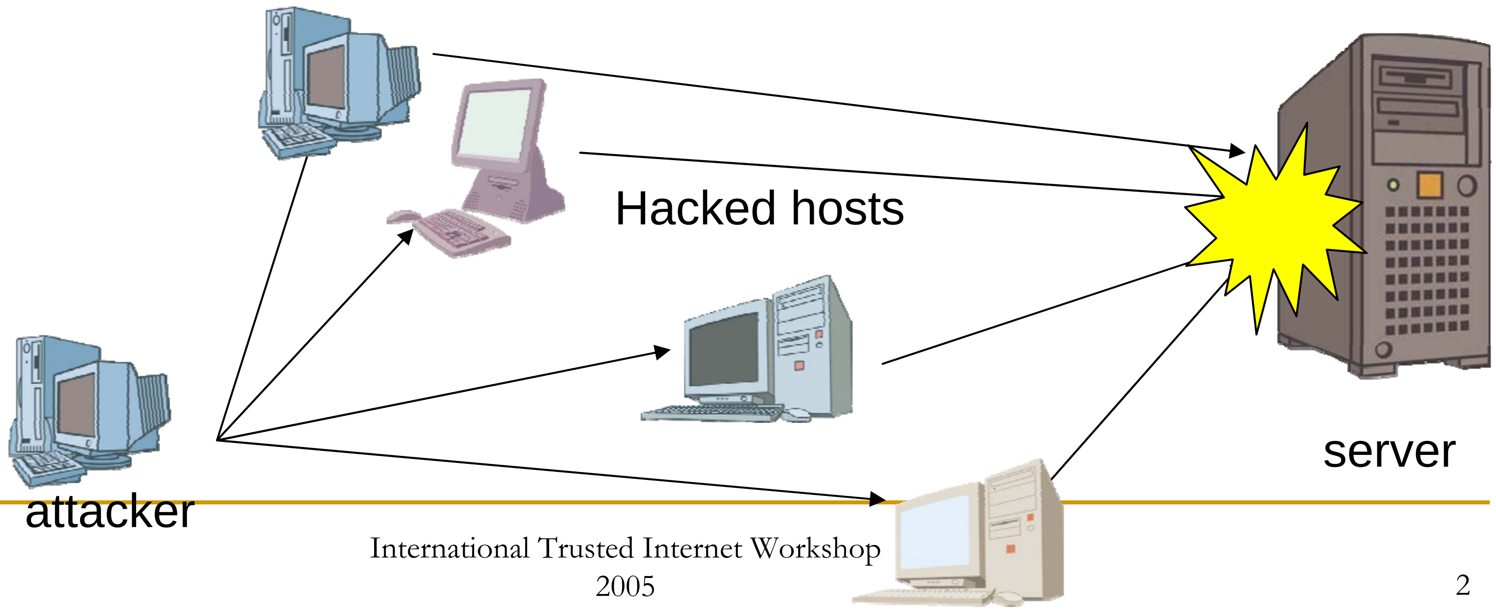
Identification of Attack Nodes from Traffic Matrix Estimation

Yuichi Ohsita
Osaka University

What is DDoS (Distributed Denial-of Service)?

- One of the most serious problems
  - Number of DDoS attacks is increasing
  - Serious economic loss
- Overview of DDoS
  - An attacker hacks remote hosts and installs attack tools
  - The hosts attack the same server at the same time



Necessity and difficulty of identification of attack nodes

- Attackers are highly distributed
  - Attacker can generate as high rate attack as a single point defense cannot deal with.

↓

- We must block attack packets at distributed points
  - To effectively block attacks, we should block on the paths from attackers to the victim.

↓

- We need to identify attack nodes
  - Problem: Identification of attack nodes is difficult
    - Attackers can easily spoof the source address

Existing methods to identify attack nodes

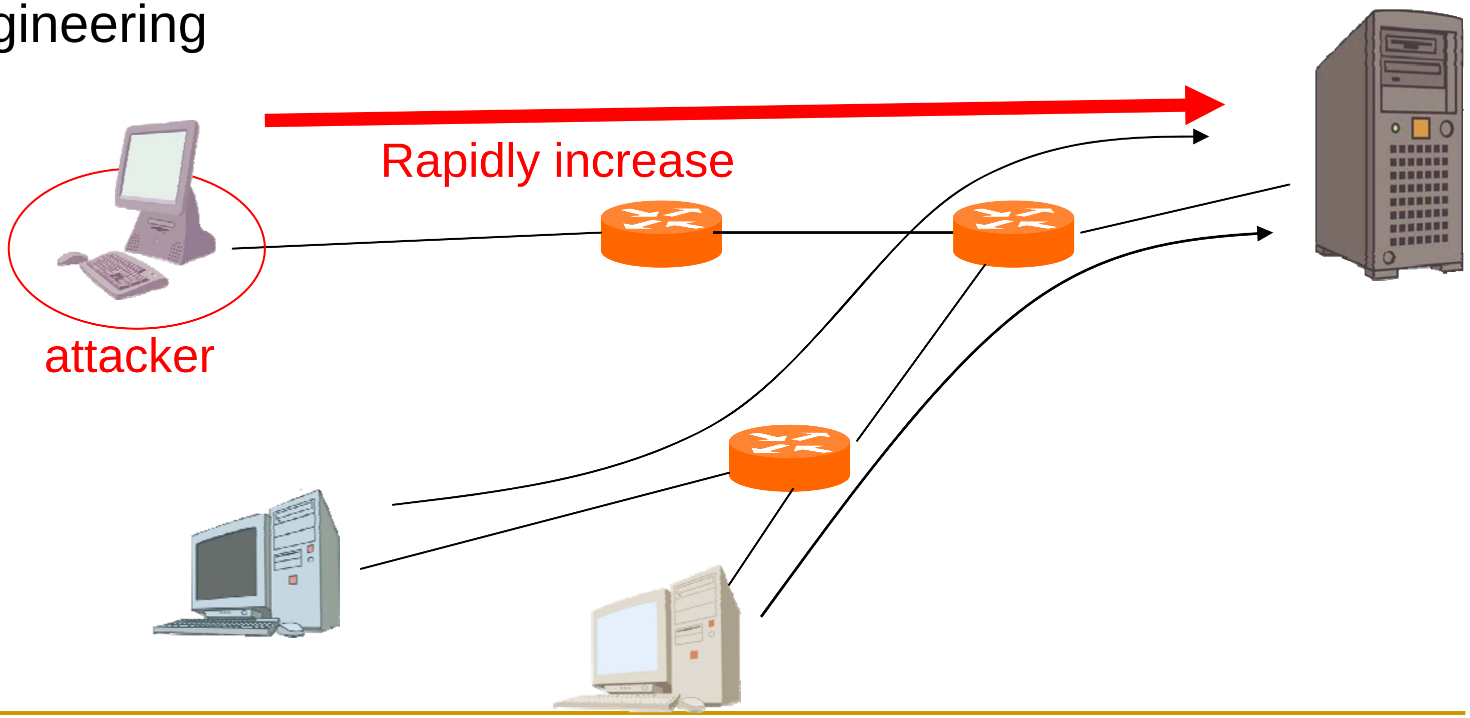
- Existing methods
  - When forwarding a packet, the router sends identification information to the destination
    - ICMP traceback, Packet Marking Method.
  - Each router stores packet digests
    - Hash-based traceback
- Problem
  - These methods cannot work with legacy routers

Goal of our research

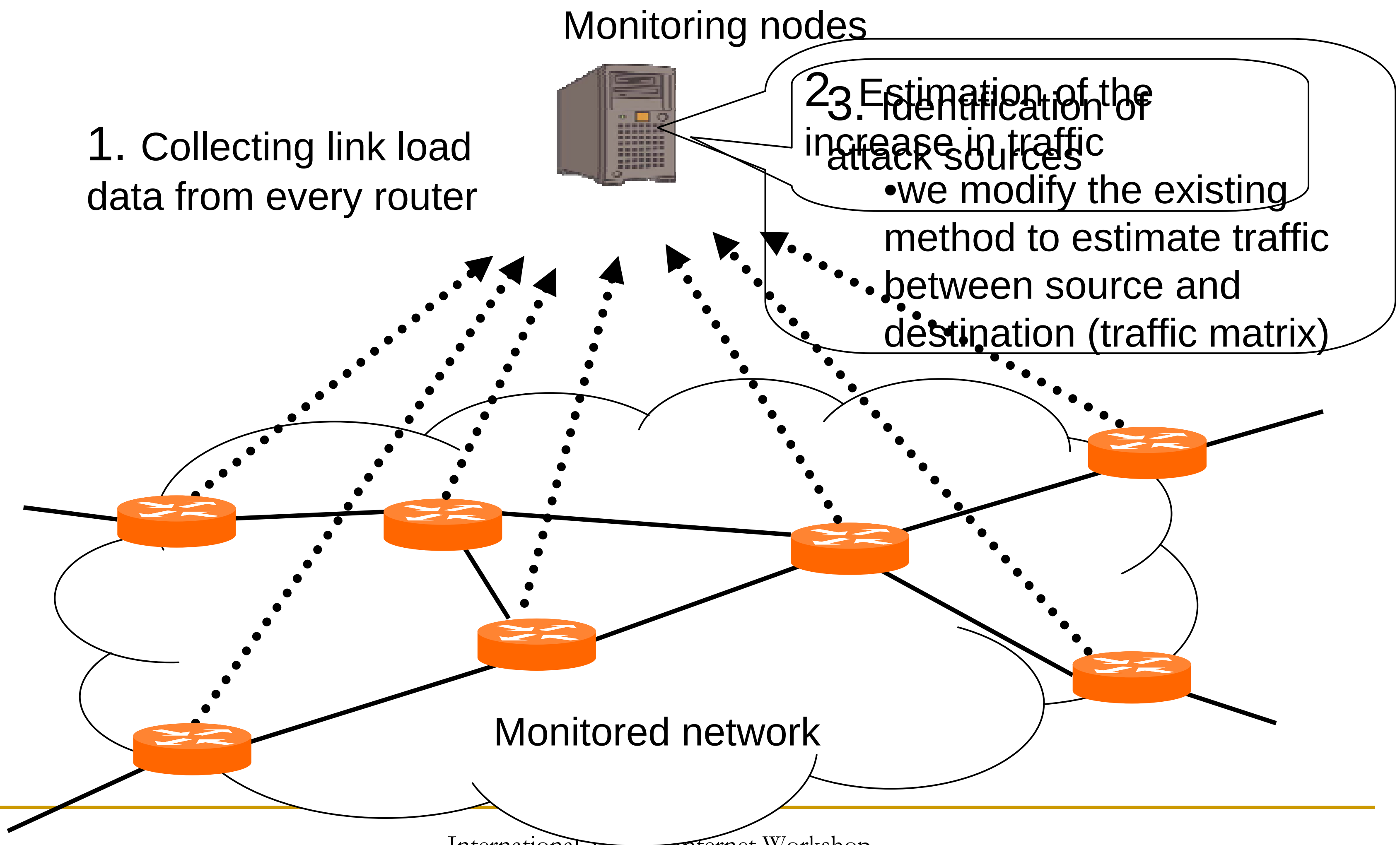
- Problem of traditional method
  - Unable to work with legacy routers
    - We must implement all or most of routers.
- Our goal
  - Identification of attack nodes which can work with legacy routers
    - Method using information which can be obtained from legacy routers
      - We can obtain statistics of link loads through SNMP

Identification of attack nodes by monitoring traffic

- We can identify the attack sources that are increasing the traffic to the victim
- Traffic between each source and each destination can be estimated from link loads by traffic matrix estimation.
  - Traffic matrix estimation is a method proposed for traffic engineering

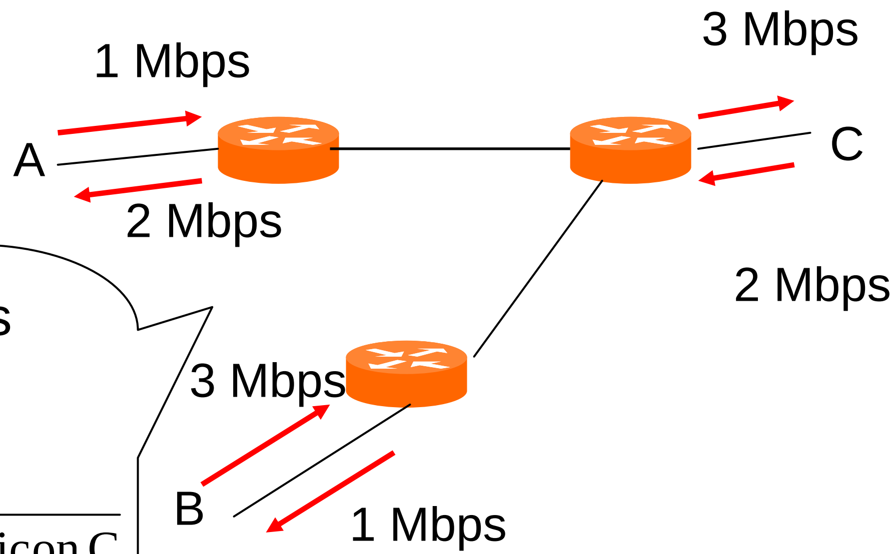


Overview of our method



Existing method to estimate traffic matrix

- Method using the gravity model
  - Typical estimation method which can estimate very fast.
  - Traffic between a source and a destination is assumed to be proportional to the total traffic at the source and at the destination.



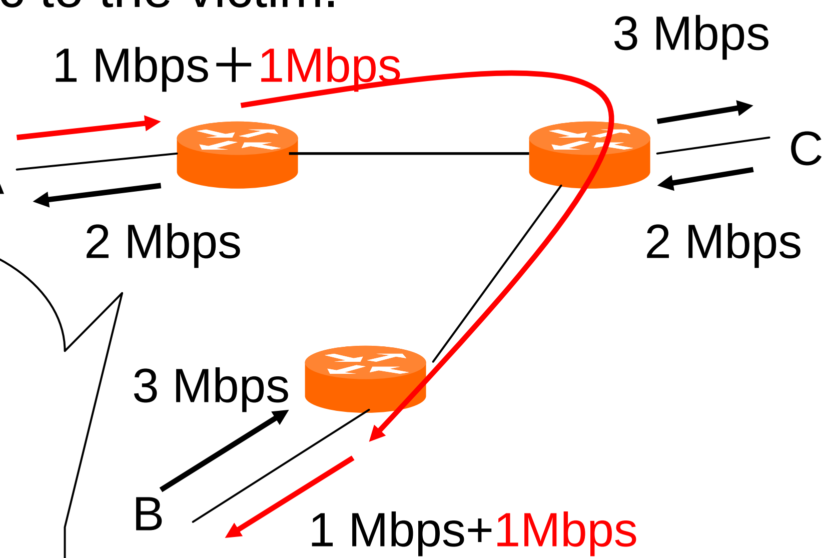
Traffic from A to B is estimated as

$$\text{Total ingress traffic on } A \times \frac{\text{Total egress traffic on B}}{\text{Total egress traffic on B} + \text{total egress traffic on C}}$$

$$= 1\text{Mbps} \times \frac{1\text{Mbps}}{1\text{Mbps} + 3\text{Mbps}} = 0.25\text{Mbps}$$

Problem of existing estimation method

- Problem of existing method using the gravity model
  - The impact of the attack traffic is distributed among the edge links that have legitimate traffic to the victim.
- Our method
  - Estimation method focusing not on the total rate but on the increase in traffic.
    - We can eliminate the effect of the amount of legitimate traffic



Even when traffic from A to B increased by 1Mbps

Traffic from A to B is estimated as

$$2\text{Mbps} \times \frac{2\text{Mbps}}{2\text{Mbps} + 3\text{Mbps}} = 0.8\text{Mbps}$$

The estimation results:

Traffic from A to B: increased by 0.55 Mbps

Traffic from C to B: increased by 0.45 Mbps

Steps to estimate the increase in traffic

- Calculation of the increase in traffic on each link

$$G = X - \overline{X}$$

G	Increase in traffic on each link
X	Loads on each link
$\overline{X}$	Average link loads of legitimate traffic

- Estimation of the increase in traffic between source and destination
  - Estimation by gravity model
  - Modification of the result by using statistics of internal links
- Estimation of the average link loads

Estimating the increase using the gravity model

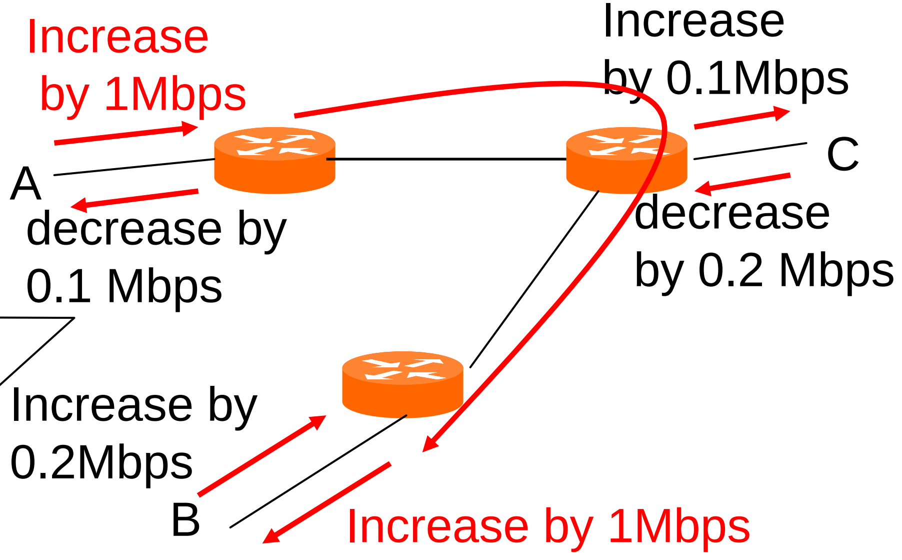
- Estimating the increase in traffic from i to j as

$$
\begin{cases}
g_i^{\text{in}} \times \dfrac{g_{\text{j}}^{\text{out}}}{\sum_{k:(g_k^{\text{out}} \geq 0)} g_k^{\text{out}}} & (g_i^{\text{in}} > 0,\ g_{\text{j}}^{\text{out}} > 0) \\
-\left| g_i^{\text{in}} \times \dfrac{g_{\text{j}}^{\text{out}}}{\sum_{k:(g_k^{\text{out}} < 0)} g_k^{\text{out}}} \right| & (g_i^{\text{in}} < 0,\ g_{\text{j}}^{\text{out}} < 0) \\
0 & (\text{others})
\end{cases}
$$

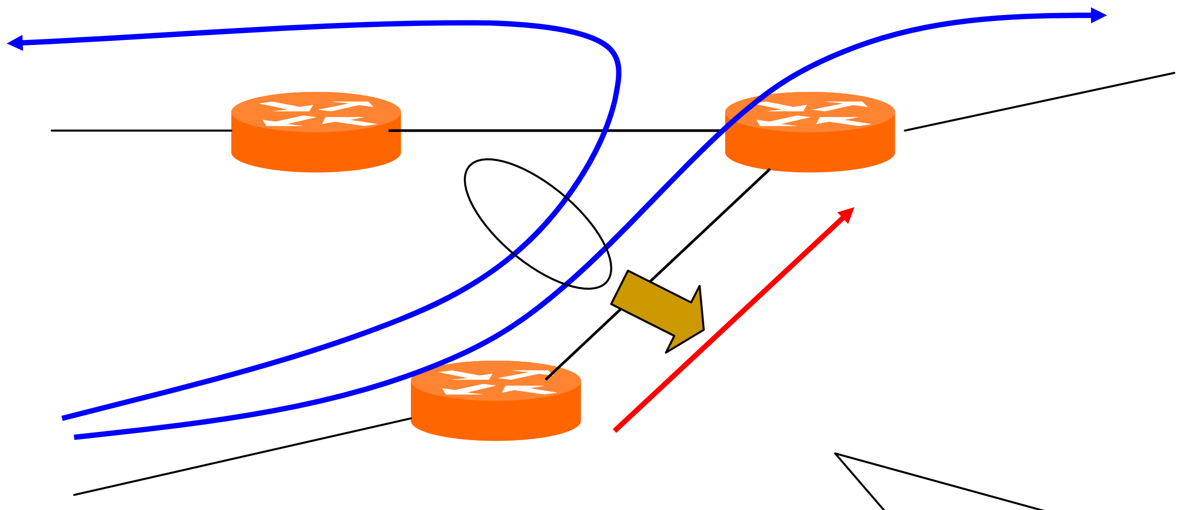
g_k^{in}	Increase in ingress traffic on link k
g_k^{out}	Increase in egress traffic on link k

The increase in traffic from A to B is estimated as

$$1\text{Mbps} \times \frac{1\text{Mbps}}{1\text{Mbps} + 0.1\text{Mbps}} = 0.9\text{Mbps}$$



Relation between link loads and traffic of flows



The total amount of traffic on the link is the sum of the traffic of flows that are passing the link

Relation between link loads and traffic of flows

- The total amount of traffic on the link is the sum of the traffic of flows that are passing the link

$$G = AF$$

F	Increase in traffic between each source and each destination
G	Increase in traffic on each links
A	Routing matrix whose entry $a_{(i,j),k}$ defined as $a_{(i,j),k} = \begin{cases} 1 & (\text{traffic from } i \text{ to } j \text{ traverse link } k) \\ 0 & (\text{others}) \end{cases}$

Using the traffic statistics on the internal links

- We adjust the increase in traffic estimated by the gravity model to satisfy $G = AF$
  - The gravity model uses only statistics on edge links

- How to adjust the increase
  - We obtain the final result F as

$$F = F' + A^{-1}(G - AF')$$

F'	Increase in traffic estimated by the gravity model
A^{-1}	Pseudo-inverse of routing matrix
G	Increase in traffic on each link

Assumption and constraint for estimating average of legitimate traffic

- We assume that the average rate of legitimate traffic $\overline{X}_n$ is basically estimated by the weighted average of the monitored traffic rate X_n

$$\overline{X}_{n+1} = \alpha X_n + (1-\alpha)\overline{X}_n \quad (0 < \alpha < 1)$$

- We must estimate the average of the legitimate traffic without the effect of sudden and rapid increase
  - This causes difficulties in the identification of the increase
- We should update the average by satisfying $G = AF$
  - Our method assumes the situation covered by $G = AF$

Steps to estimate average of legitimate traffic

- We extract the element not increasing rapidly from estimated traffic
  - We define $\hat{F}_n$ as a vector whose element $\hat{f}_{(i,j)}$ is
    - 0, in the case that traffic from i to j increase rapidly
    - Otherwise, the estimated increase in traffic from i to j
  - We can eliminate the effect of rapid increase
- We update $\overline{X}_n$ as

$$\overline{X}_{n+1} = \alpha(\overline{X}_n + A\hat{F}_n) + (1-\alpha)\overline{X}_n \qquad (0 < \alpha < 1)$$

    - A is the routing matrix
  - We can update the average by satisfying $G = AF$

Assumption for identification of attack nodes

- Attack nodes are the sources increasing the traffic on the victim
  - When an attack starts, the traffic sharply increases from the attackers to the victim.
  - The larger the increase is, the more serious the impact on the network resources is.
- The total rate of attack traffic can be estimated from the increase of the egress traffic to the victim.
  - Setting a static threshold to the increase in traffic is not sufficient.
    - When the number of attackers is large, the impact is serious even if the rate from each attacker is not so large.

Steps to identify attack nodes

- Estimate total attack rate $\tilde{g}^{\text{out}}$

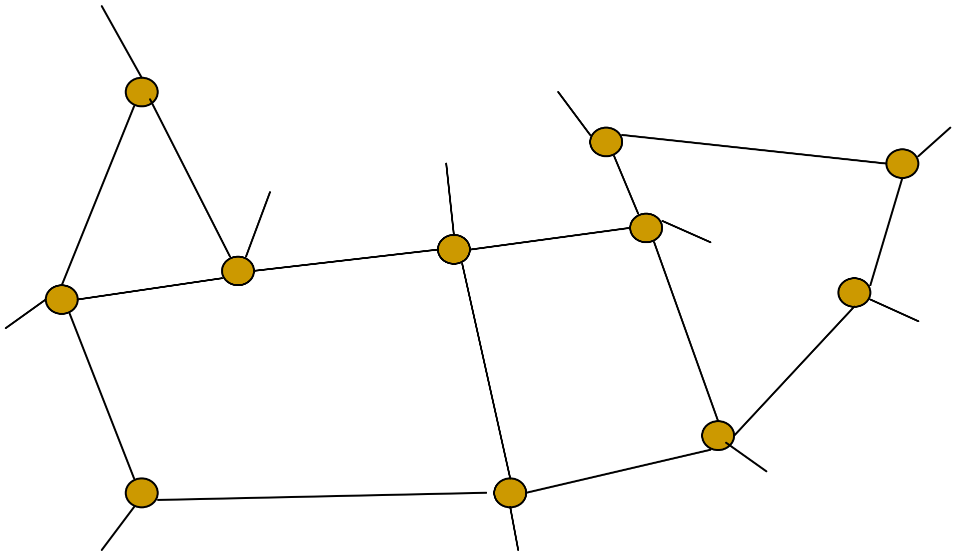
$$\tilde{g}^{\text{out}} = g^{\text{out}} - \mu^{\text{out}} - \gamma$$

g^{out}	Increase in traffic on the link connected to the victim
μ^{out}	the average of the last J values of g^{out}
γ	parameter indicating the variation in the rate of the legitimate traffic

- We identify the source of the largest estimated increase as attack source
  - The identification of another attack node is continued until the sum of estimated increase of identified attack nodes is larger than $\tilde{g}^{\text{out}}$

Evaluation

- We evaluate our method by simulation
  - Topology
    - The backbone topology of Abilene
  - Legitimate traffic pattern
    - Traffic monitored at the gateway of Osaka University
      - We made 110 groups of packets based on a 16 bit prefix of the source address.
      - We calculated the aggregated traffic rate for each group at a 60 seconds interval.



Metrics used for evaluation

- False-positive
  - Cases where a source not generating attack traffic is erroneously identified as an attack source.
- False-negative
  - Cases where an attack source cannot be identified.
- False-positive rate
  - $\frac{\#\text{ of false-positve}}{\#\text{ of attack nodes}}$
- False-negative rate
  - $\frac{\#\text{ of false-negative}}{\#\text{ of sources not generating attack traffic}}$

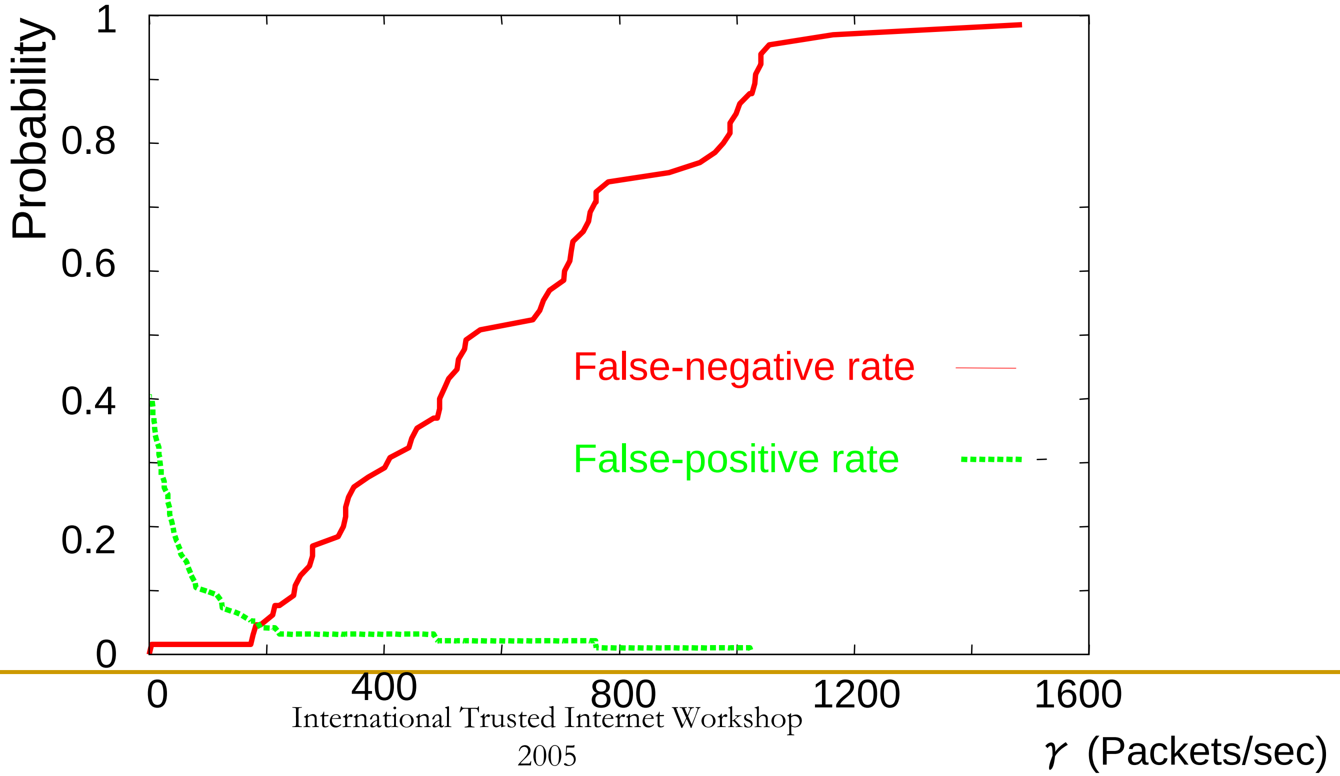
Number of attack nodes vs. false-positive, false-negative

- We simulate attacks changing the number of attack nodes from 1 to 5
- We injected attack packets at 16 different times.
- The total rate of attack traffic is 1000 packets/sec irrespective of the number of attack sources
- We set γ to 200 Packets/sec
- Our method can accurately identify attack sources regardless of the number of attack nodes

# of attack nodes	# of False-negatives (false-negative rate)	# of False-positives (false-positive rate)
1	0 (0.00)	2 (0.01)
2	0 (0.00)	0 (0.00)
3	0 (0.00)	3 (0.02)
4	3 (0.04)	4 (0.04)
5	12 (0.15)	4 (0.05)

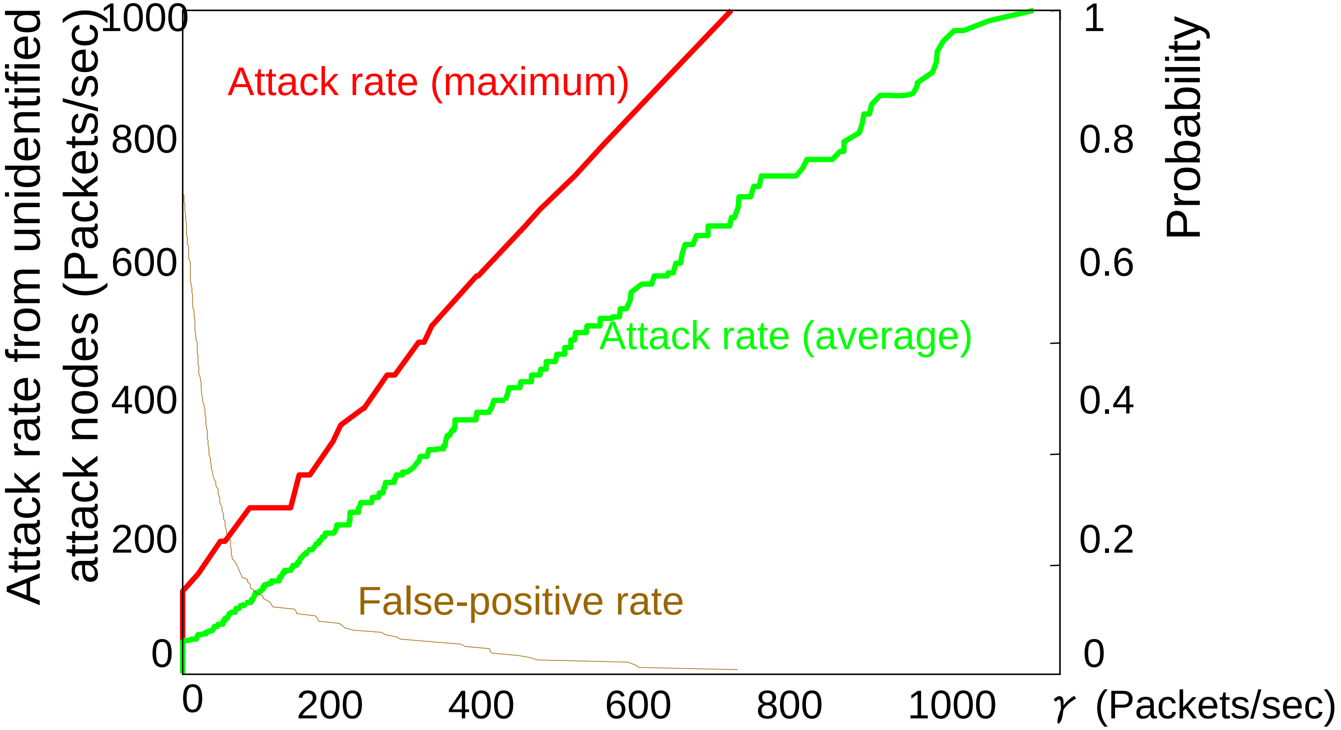
γ vs. false-positive, false-negative

- Our method can reduce the number of false-positives by setting γ to a larger value.
- A large γ causes many false-negatives.



γ vs. attack rate from unidentified attack nodes

- We simulated our method, changing the attack rate.
- We injected attack packets at 16 different times.
- Number of attack nodes is 4.
- The total rate of attack traffic from unidentified attack sources is closely related to γ
  - We can set γ adequately by defining the maximum attack rate that does not affect the network resources.



Conclusion

- We propose a method to identify attack nodes by estimating the increase in traffic between sources and destinations
  - Our method can work with legacy routers
    - The increase is estimated from link loads which can be obtained through SNMP
  - Our method can distinguish attack nodes from legitimate clients
    - We use the increase to identify attack nodes
- Simulation results show that our method can accurately identify attack nodes