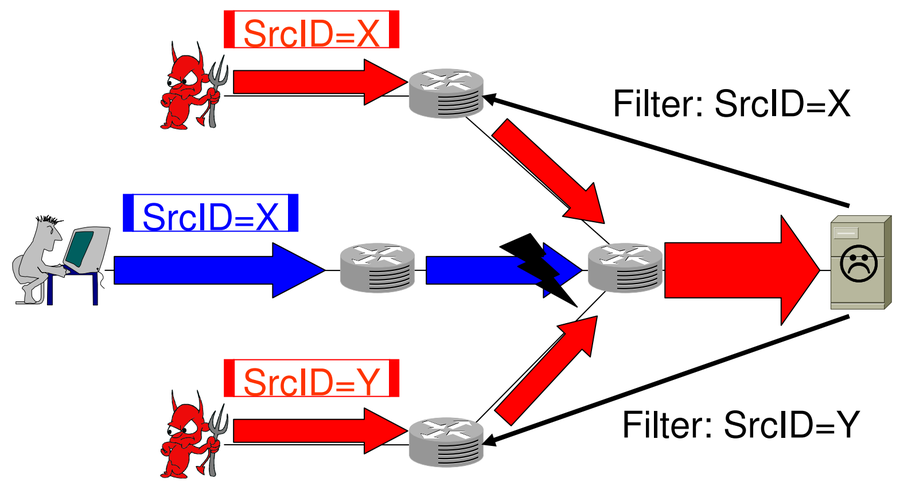
Unspoofable Source Identifiers without Global Trust

Xin Liu, Xiaowei Yang, David Whetherall, and Thomas Anderson

Motivation



Benefits:

- Providing unspoofable packet signatures for filtering
- Providing unspoofable identifiers for fair resource allocation
- Limiting reflector attack
- Useful for deterring future attacks

Solution: Packet Passport

IP Packet

IP Header	**Passport**	Payload

Every packet carries a passport, and packets with invalid passports are dropped or demoted.

Key Features

- An attacker **cannot forge** valid passports.
- It is **efficient** to generate and verify passports at packet forwarding time.
- The passport system is **scalable** and **robust** against DoS attacks.
- The passport system supports **incremental deployment** and provides **incentives for early adoption.**

Previous Solutions to Packet Source Identification

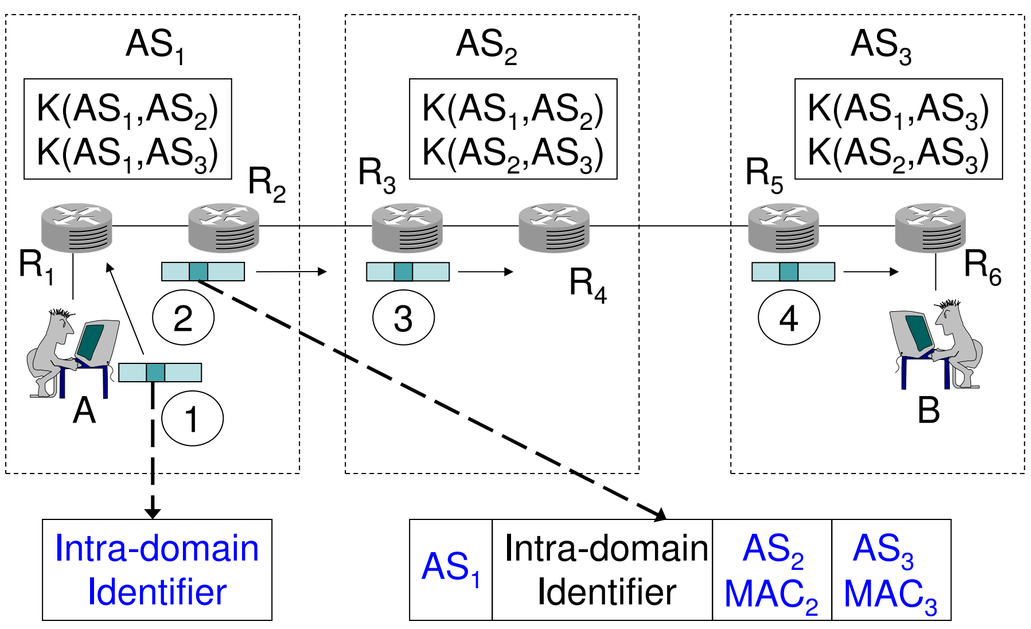
Fast but weak:

- *Ingress/egress filtering*: Source address is not verifiable.
- *Path Identifier [Yaar03] & AITF [Argyraki05]*: Part of a path identifier is spoofable.
- *Authenticated Marking Scheme [Song01]*: The path identifier is not verifiable at packet forwarding time.
- *Spoofing Prevention Method [Anat05]*: The secrets are transmitted in plain text, and there is no secret exchange protocol.

Strong but inefficient and unscalable:

- *R.Perlman's PhD Thesis [Perlman88]*: Public key signatures are in packets to identify their sources.

Passport Processing



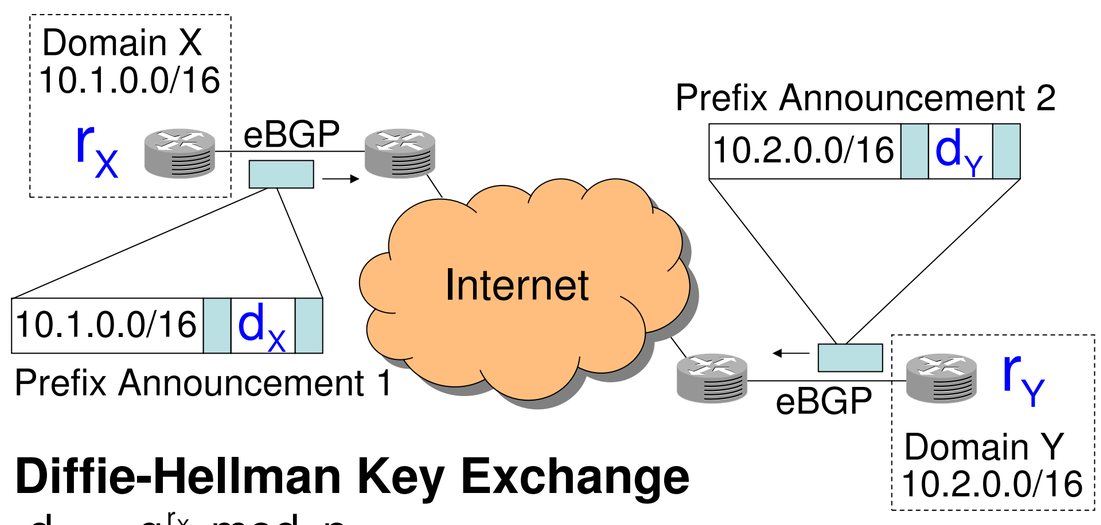
MAC: Message Authentication Code

MAC_X $= MAC_{K(AS_1,AS_X)}(AS_1, AS_2, AS_3, SrcIP, DstIP, \cdots)$

K(X,Y): Symmetric key shared between two domains X and Y

1. Intra-domain identifier is inserted.
2. Intra-domain identifier is verified & full passport is inserted.
3. MAC_2 is verified using the key $K(AS_1,AS_2)$.
4. MAC_3 is verified using the key $K(AS_1,AS_3)$.

Key Distribution



Diffie-Hellman Key Exchange

$d_X = g^{r_X} \bmod p$

$d_Y = g^{r_Y} \bmod p$

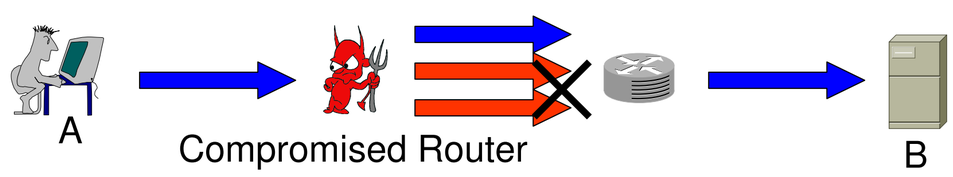
$K(X, Y) = d_X^{r_Y} \bmod p = d_Y^{r_X} \bmod p$

Preliminary Evaluation

Practical with today's hardware technology

- Passport generation & verification: with UMAC, a commodity PC can generate 975K passports and verify 3.9M passports per second.
- Key distribution: computation, communication and storage costs are negligible.
- Bloom filter: 16MB SRAM can "remember" 2.5Gbps traffic for 5 seconds with a false positive rate of 5.7×10^{-6}.

Preventing Replay Attack



Problem: Attack sources cannot be identified.

Solution:
- Using *bloom filters* to detect passport duplication
- Using *fast-rekeying* to deal with bloom filter flushing

For details, please visit:

http://nds.ics.uci.edu/pktpassport