Anonymous DTN routing

1 Introduction

In this project, we propose a routing protocol which provides *identity anonymity* and *location anonymity* against untrusted participants in DTN. Identity anonymity means that the identity of packet sender or destination is not revealed to any other nodes whom the sender or destination don't trust. Location anonymity implies that the geographic location of a node cannot be tracked by other nodes whom the node doesn't trust. In addition, the proposed protocol should show reasonable efficiency in terms of packet delivery rate, bandwidth and packet delivery latency.

The protocol achieves sender identity anonymity and weak receiver identity anonymity against untrusted nodes through the use of ephemeral ID. Each node i in DTN generates and updates its own ephemeral ID, and other nodes that i trusts are able to generate the current ephemeral ID of i at any time. However, attackers or nodes that i does not trust cannot learn the permanent ID of i. The attackers may receive the current ephemeral ID of i from i or other nodes, but they cannot generate ephemeral ID of i on their own. Therefore, even though the attackers observe or receive packets sent by i, they cannot determine the permanent ID of i. Moreover, the attackers recognize packets from a single node over different epochs as several streams of packets from several nodes, since the ephemeral ID of the sender is changed periodically.

Unlinkability is also achieved by the use of ephemeral ID. Unlinkability may be broken if packet delivery from a sender to a destination is completed within a single epoch, that is, ephemeral IDs of the sender and destination are not changed during the delivery. Even in this case, attackers are only able to learn ephemeral IDs of the sender and destination and cannot map the ephemeral IDs to the permanent IDs. Moreover, the length of epoch is set to be reasonably short so that ephemeral IDs of sender and destination contained in a packet would be changed at least once during the delivery.

Location anonymity is achieved by the use of ephemeral ID which is changed periodically. Since any kind of analog finger printing is not considered in this project, attackers cannot track a node without knowing a permanent ID and a random seed of a victim node that are used for generating ephemeral ID.

2 Protocol Design

2.1 Network initialization

On entering a DTN network or at the end of each epoch, a node i builds a set of ephemeral IDs of its trusted nodes. An ephemeral ID of i is generated using a hash function which takes the permanent id, a random seed and time as inputs. Then i updates ephemeral IDs of its trusted nodes and the destination addresses of the packets stored in i. i maintains the recent contact history consists of untrusted nodes that i has met and resets the history at the end of each epoch since it cannot generate new ephemeral IDs of untrusted nodes.

2.2 Connection setup between two nodes

After network initialization, a node j advertises its presence to node i by generating a beacon message with an ephemeral ID of j. Upon receiving a beacon message from j, i may send j a hello message which contains a packet digest of i, that is, a bloom filter encoding of ephemeral IDs of destinations of the packets stored in i. Since the packet digest contains ephemeral IDs of destinations of packets, j can calculate an intersection of 1) the packet digest and 2) its own trusted nodes the recent contact history, and send i a pulling message with the calculated intersection.

2.3 Packet construction

A packet sender constructs a packet by first encrypting the message along with its ephemeral ID using a symmetric key shared between the sender and destination of the packet. Then the sender generate a packet which contains the ephemeral ID of the destination and the encrypted message.

2.4 Packet forwarding

A node i has a packet to relay, or has generated a packet and wants to send it to destination node. Then whenever i contacts j and receives a pulling message from j, i forwards the packets pulled by j.

If j is the destination node of the packet, i insert the packet into its TX queue and set the highest priority for the packet. Then i removes the packet from its RX queue. If j is not the destination but trusted by the destination, then i inserts the packet into the TX queue with the second highest priority and keeps the packet for further replication. If j is neither the destination nor a node trusted by the packet destination, i insert the packet into the TX queue with the lowest priority. Those packets with the lowest priority are sent to j only after all the other enqueued packets with higher priority are forwarded to j.

3 Simulation

Currently we are implementing the anonymous DTN routing protocol using *ONE simulator*. ONE simulator supports a few existing DTN routing protocols such as epidemic, MaxProp and Prophet. We will implement a standalone anonymous DTN protocol and also modularize our implementation so that it can be embedded into the existing DTN routing protocols.

Current status is as follows:

- Network initialization: Done
- Connection setup between two nodes: On-going
- Packet construction
- Packet forwarding