

Secure Peer-ID Assignment in P2PSIP

Eric Rescorla

Network Resonance

`ekr@networkresonance.com`

Background

- Need to be able to authenticate that a peer has a given ID
 - Otherwise a variety of routing attacks are possible
- In practice, this means cryptography
 - Need to bind peer-id X to its public key
 - But how?

Cryptographically Generated Peer-Ids

- Peer generates a random key pair K_{pub}, K_{priv}
 - $I = SHA1(K_{pub})$
 - This gives you a “random” peer-id
 - * Because of the SHA-1
- How does authentication work?
 - Peer signs something with K_{priv} and sends *signature*, K_{pub} , I
 - Relying party verifies signature and that $I = SHA1(K_{pub})$
- This is the technique used in HIP

Chosen Location Attacks

- Attacker wants to get between X and $predecessor(X)$
 - A random node-id has a $1/N$ chance of being in $(pred(X), X)$
 - * Where N is the number of nodes in the overlay
 - * The size of the hashspace is irrelevant
- An attacker can succeed in average $N/2$ trials
 - This is an offline attack
- Two basic countermeasures
 - Slow down the search (but keep it offline)
 - Make it an online attack

Proof of Work

- Idea: make generating candidates expensive
 - Example: partial preimage
 - * Joining peer generates random X
 - * $PeerId = SHA1(X)$
 - * Bottom n bits of $PeerId$ must be random
 - Need to try average 2^{n-1} X values to get a valid $PeerId$
 - * This increases search cost by 2^{n-1}
- The puzzle must be tied to the peer-id
 - Otherwise the attacker can solve the puzzle once and then generate many peer-ids
 - This is why CAPTCHAs are hard to deploy here
- This only works well when the attacker isn't powerful
 - Not true with botnets

Invitations

- What if an existing peer asks you to join [MI07]
 - You start as a client with peer-id X
 - * But you can't attack anyone since you're not a peer
 - The responsible peer invites you to become a peer
 - * Splits his zone of responsibility with you
- Not clear how this helps
 - Attacker chooses his peer-id
 - * Joins the overlay
 - * Waits to be invited as a client
 - Also, how do you cryptographically bind key to peer-id

Central Enrollment Server

- We have a central server
 - Joining peer contacts the server with his public key
 - Server validates peer somehow
 - Server issues a certificate with a random peer-id
- This makes the attack online
 - Even if no authentication is performed, you need a lot of queries to the server
 - If you have user authentication, then you only get one query