TURN
draft-ietf-behave-turn-11

Philip Matthews
Rohan Mahy
Jonathan Rosenberg
Recent work on TURN

- Two versions since Dublin: TURN-10 and TURN-11
- TURN-11 currently in WGLC
Changes: -09 to -11 (1 of 8)

- Added CreatePermission transaction
  - Permissions can only be created/updated by an authenticated transaction (CreatePermission or ChannelBind transactions)
  - Send indication and ChannelData messages no longer affect permission
  - Must create a permission before sending to a peer, to avoid attack found by Cullen
  - Can create/update multiple permissions in one CreatePermission transaction
    - Reduces the overhead of managing permissions.
Changes: -09 to -11 (2 of 8)

• Added DONT-FRAGMENT attribute
  – Include in Send indication to request that DF bit be set when sending to peer
  – Include in Allocate request to test whether server can support
  – Allows a limited form of Path MTU Discovery
Changes: -09 to -11 (3 of 8)

• Changed how ALTERNATE-SERVER attribute is used
  – Must only use with error code 300 (Try Alternative)
    • Previously allowed with other error codes
  – May appear in unauthenticated responses
    • Previously, only allowed in authenticated responses
  – Allows anycast discovery of TURN servers
Changes: -09 to -11 (4 of 8)

• Removed concept of preserving allocations
  – Simplified document; concept can be added later

• Replaced REQUESTED-PROPS with EVEN-PORT
  – Can request an even port number, or an even port number with next higher port number reserved
  – Removes concept of “flags” for future extensions
    • Now, only way to signal a new feature is through a new comprehension-optional attribute
Changes: -09 to -11 (5 of 8)

- Reduced the range of channel numbers
  - Now: 0x4000 through 0x7FFF
  - 0x8000 through 0xFFFF reserved
    - Allows for future extensions
- Allow 508 responses for any capacity problem
  - Allows a server to fail attempts to create a new permission or channel due to memory constraints, etc.
Changes: -09 to -11 (6 of 8)

- Corrected “SOFTWARE-TYPE” to “SOFTWARE”
  – Recommended only in Allocate and Refresh transactions, though can be used elsewhere

- Renamed attributes: XOR-PEER-ADDRESS and XOR-RELAYED-ADDRESS

- Minor changes to semantics of Allocate, Refresh, and ChannelBind to support idempotency over UDP transport.
Changes: -09 to -11 (7 of 8)

- Allow server to restrict range of addresses and ports that can be specified as a peer.
- Client must now wait 5 minutes after a channel binding expires before reusing the channel number or the transport address in another binding.
- Recommended that the server impose quotas on the number of allocations and the bandwidth used by given username at any one time.
Changes: -09 to -11 (8 of 8)

• Removed all support of IPv6 to TURN-IPv6
• Added a long and detailed example
  – Includes attribute values
• Various other minor changes. See section 21 in the document for details.
Issue 1: TURN server names

- Spec says:
  - By default, TURN runs on the same port as STUN
  - TURN uses new SRV service names "_turn" and "_turns"

- Can IANA handle this? Can one allocate a service name without allocating a new port number?
Issue 2: TURN Loop Attack

Allocate request. Src = B’s transport address
Issue 2: TURN Loop Attack

Allocate response.

Critical assumption: attacker can see response
Issue 2: TURN Loop Attack

Allocate request. Src = B’s transport address
Issue 2: TURN Loop Attack

Allocate response.

Critical assumption: attacker can see response
Issue 2: TURN Loop Attack

ChannelBind request.
src = server B’s transport addr
peer = server B’s relayed addr
Issue 2: TURN Loop Attack

TURN Server A

TURN Server B

ChannelBind request.
(Like before, but replace B with A)

Attacker
Issue 2: TURN Loop Attack

TURN Server A

Attacker

TURN Server B

ChannelData message.
Issue 2: TURN Loop Attack

```
TURN Server A

Attacker

TURN Server B

Loop
```
Issue 2: TURN Loop Attack variant

TURN Server A

TURN Server B

Loop

Attacker
Issue 2: TURN Loop Attack

**Assumptions:**
- At least one server does not decrement the TTL
- If both servers use authentication, then attacker needs to be able to see Allocate responses even through they are addressed to the other TURN server.
  - If no authentication, then attacker might be able to guess the allocated relayed-transport-address

**Proposal:**
- Mention this attack in document
- This is another reason servers should use authentication
- Document already says that servers should decrement TTL if they can
- Don’t do anything more
Finally …

• Please read and comment during WGLC period!